Faculty of Computer Science & Information Technology
University Of Malaya
50603 Kuala Lumpur
Malaysia

*PROJECT TITLE*

# VPN- DATA CENTER SYSTEM DATABASE

*Presented by*
Gan Thai Hwan (WEK 98021)

*Supervised by*
Pn. Miss Laiha

*Moderated by*
Pn. Hannyzzura

*Dissertation submitted by in partial fulfillment of the requirements for the Degree of Bachelor of Computer Science*

## Abstract

Computer has advanced from high technology equipment that only by large company in the old days to office equipment that is essential in an office. In the world of computer, improvement on service provided and lock out competition is very important. The quantity of service, speed of processing, productivity, efficiency and effectiveness is very vital.

The purpose of this project is to create a database system to a company that provides services and selling computer product. Beside that, we also provide VPN-Virtual Private Network for the company's branch to access and to update the database. This is to provide more security to the data while reducing the traditional cost of Remote Access Server (RAS) connections. Other than that, this project also provides the online purchase and services by only the limited customer. This is easier than the traditional way, which is by telephone or manually.

This project is using the VPN server to configure the tunnelling and the encryption and description of the data. The database is created using Microsoft Access and the data entry form to access to the database is created using Visual Basic 6.0.

## Acknowledgement

First of all, I would like to thank Pn. Miss Laiha, my supervisor for giving me to join this project and provide me the fully equipment. Secondly, I would like to thank her for his constructive advice, generous guidance, encouragement, support and supervision along the project. His hardworking and kindness in helping me throughout the project is deeply appreciated.

I would also like to thank my modulator, Pn. Hannyzzura for her advice to help me develop the processing for proposal system.

I want to express my gratitude to my teammate Mr. Wong See Wai and Ms. Tan Lean Lee who have helped me and given support and assistance to me throughout the project. Thanks to my entire course mate for their valuable ideas and help.

Last but not least, I would like to thank the technical staff of the Faculty of Computer Science and Information Technology, University of Malaya for their technical advice and assistance throughout my project.

## Table of Contents

## List of Figures

## List of Tables

# CHAPTER 1 INTRODUCTION TO THE PROJECT

## 1.1 Overview

In the world of the technology, improvement and development never stop. Computer has advanced from high technology equipment that only by large company in the old days to office equipment that is essential in an office. In the world of computer, improvement on service provided and lock out competition is very important. The quantity of service, speed of processing, productivity, efficiency and effectiveness is very vital.

A VPN is a communications network, built for the private use of the enterprise, over a shared public infrastructure such as Internet. There are two primary applications covered by this definition: remote access connectivity and site-to-site connectivity. Three primary applications of VPN covered by this definition are remote access VPN, Intranet VPN and Extranet VPN.

***Intranet:*** VPN facilitate secure communications between a company's internal departments and its branch offices. The primary technology requirements are fast, strong encryption to accommodate the high-speed links present in internal LANs; reliability to ensure the prioritization of mission-critical applications, such as financial systems, sales and customer database management, and document exchange; and ease of administration to accommodate the changing requirements of new users, new offices and new applications.

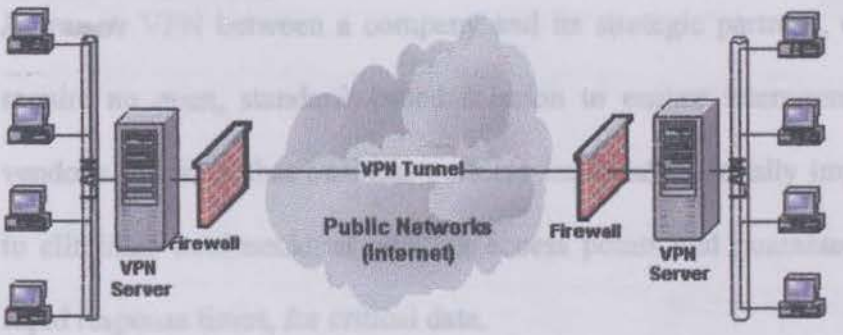


**Figure 1.1: Intranet VPN**

***Remote Access:*** VPN between a company and its remote and/or mobile employees have different requirements. Reliability and Quality of Service are important, because the employees accessing the VPN are typically limited to slow modem speeds. Additionally, strong authentication is critical to ensure the remote and mobile users' identities in the most accurate and efficient manner possible. On the management side, Remote Access VPN require centralized management and a high degree of scalability to handle the multitude of VPN links, as well as the vast number of users accessing the VPN.

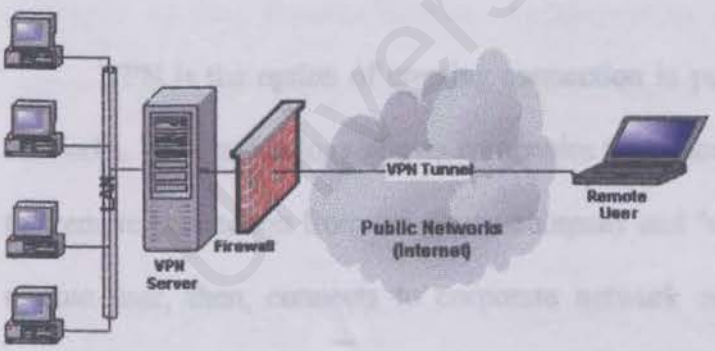


**Figure 1.2: Remote Accesses VPN**

***Extranet:*** VPN between a company and its strategic partners, customers and suppliers require an open, standards-based solution to ensure interoperability with the various vendor's solutions that business partners implement. Equally important is traffic control to eliminate bottlenecks at network access points and guarantee swift delivery of, and rapid response times, for critical data.

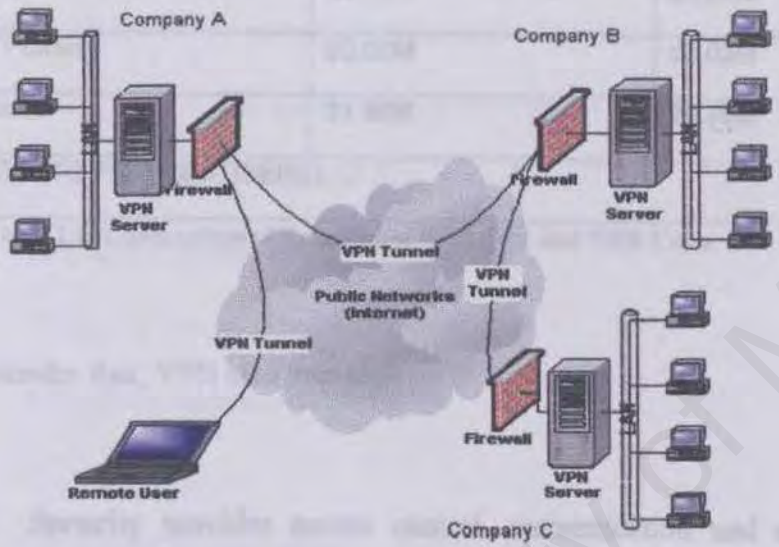


**Figure 1.3: Extranet VPN**

VPN is the option of creating connection in purpose to reduce cost of enterprise networks. VPN technology allows companies to reduce remote access costs by removing the remote connection from the phone company and "outsourcing" it to the Internet. The remote user, then, connects to corporate network resources through a local Internet Service Provider (ISP) rather than over expensive phone lines. The table below shows the comparison of the traditional cost of Remote Access Server (RAS) connections versus today's Internet-based VPN.

| | Traditional RAS Costs | VPN Costs |
|---|---|---|
| Phone/ISP Charges | $1.08M | $0.54M |
| User Support | $0.30M | $0.00M (included in user access costs) |
| Capital Expenses | $0.10M | $0.02M |
| T1 Lines | $0.02M | $0.03M |
| **Total** | **$1.50M** | **$0.59M** |
| (COSTS PER 1000 USERS) | | |

**Table 1.1: Comparison of Traditional RAS Cost and VPN Costs**

Besides that, VPN also provides:

- **Security** provides access control, authentication and encryption technologies to guarantee the security of network connections, authenticity of users, and privacy and integrity of data communications. Authentication is the most important security element of a multiprotocol IP tunnel. Authentication ensures that tunnels will only be established between verified tunnel partners.

- **Reliability** provides bandwidth management, Quality of Service, and high-speed performance to guarantee the reliability, and performance of the VPN.

- **Enterprise Management** provides integration of VPN with enterprise security policy, local or remote centralized management, and scalability of the solution.

## 1.2 Objective

The general objective of project is to establish remote access VPN connection and to develop a Data Center System through this connection. This Data Center System is developed in two parts, user application and web application. The purpose of developing this system is to retrieve data from a Data Center Server. The fundamental objective of the VPN- User Application Data Center System is

- to develop a database and data entry form
- to provide added security to the data
- to update the database remotely
- to capture the status of stock

## 1.3 Project Expectation

The system enable frequent update of the database, databases in the system should be synchronised. The databases will be synchronised whenever the user login to the page. With this, the VPN- Data Center System should be able to:

- Provide an environment consists of databases.
- Provide a platform for future expansion of the system.
- Ensure integrity of data inside all database and flexible function
- Provide secured and stable access to all database.
- Well-organised and integrated system.
- Provide a user friendly environment to the user

## 1.4 Project Scope

We have stated four tasks in our project. Here are the tasks:

- VPN Establishing

In this task, VPN connection will be established between data center server and remote access computers in secured manner.

- User Application

This is window-based application that retrieving data from data center. It will contain few data entry form. It also can be used to generate report. This application will be written with Visual Basic.

- Web application: data retrieving form

Similar with the client application, this is web-based application that also used to retrieve data from data center. Data can be edited and upgraded by multiple users with the data entry forms in web pages.

- Web application: report generating

Data can be queried from the data center by using web pages and appropriate report is generated in form of html.

My task is to create a User Application and my scope is written below:

- To develop a database at the main company to keep data.
- Create data entry form for user to access the database from the main server.
  - This data entry form is only for the branches of the company.
- Generate reports from the database.
- Design and create a query to enabled user to search appropriate data from database.
- Develop a tool to allow the user manages and transfers inventory from branch to store and via versa.
- Create a tool for user to manage record of business such as user record and product record.

## 1.5 Project Scheduling

Project scheduling cursors of the whole development activities is carefully planned out to achieve a systematic progress and ensure on-time delivery of the product. It is important to have a project schedule as it acts as a time management and control to the developer making sure he is in route of the direction of the project. The project schedule is as shown below:

| Project Schedule | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| **Task** | **Start** | **Finish** | **June** | **July** | **Aug.** | **Sept.** | **Oct.** | **Nov.** | **Dec.** | **Jan** |
| 1. Research | 15/6/2000 | 10/8/2000 | | | | | | | | |
| 2. Requirement & Analysis | 1/7/2000 | 7/9/2000 | | | | | | | | |
| 3. System Design | 1/9/2000 | 31/10/2000 | | | | | | | | |
| 4. Modules Development | 15/9/2000 | 7/12/2000 | | | | | | | | |
| 5. System Integration & testing | 1/12/2000 | 31/12/2000 | | | | | | | | |
| 6. Documentation | 1/9/2000 | 15/1/2001 | | | | | | | | |

* Modules Development includes module design, coding, prototyping and testing.

**Figure 1.4: Project Schedule**

# CHAPTER 2 LITERATURE REVIEW

## 2.1 Project Studies

Before a project is being developed, it needs a lot of studies and analysis. Even many companies have moved to e-business, there are still number of them process the transaction locally.

For example inventory control system, some enterprises need to log in into the system locally every day to update the stock level. Since that is a stand-alone computer, which didn't have any network connection, administrator (person-in-charge) has to go to the computer to copy out the data and bring to another computer to print the output. To generate the report, every record has to be reviewed and record manually. This creates a lot of unnecessary jobs and much time needed to generate a complete record. However, if the system can straight away create the report with just a few clicks, it will surely speed up the whole process and the persons-in-charge will be able to do other jobs that are more productive.

## 2.2 List of Resources Surveyed

The various resources that were surveyed can be summarized as research that was done in order to obtain and gather reliable, relevant, adequate and comprehensive information. This information gathering was done by using a number of methods such as through the Internet, lecturer supply materials and also ideals, reading from newspaper, brochures, journals, articles, books and other relevant materials.

**Surfing the Internet**

Internet surfing in today's world is a very efficient way of gathering information. There are many web-sites available that provide useful and expertise information, which is needed in this system. Some web-sites have provided very useful samples that can serve as guidelines in determining system requirements.

**Research**

Research involves reviewing books and journals that contain relevant information. Researches also include gaining knowledge from friends about VPN and information about creating a database system.

**Discussion**

Discussion with group members once a week is very important to gather information and ideas. Advice and guidance from my supervisor, Puan Miss Laiha, is very important in developing a systematic system. She also helps me to solve problems and giving me some ideas during the process of developing the system.

## 2.3 Authentication

### 2.3.1 Password and Password Usage

A password is a sequence of characters that can be used for several authentication purposes. Passwords are often used to authenticate the identity of an automated data processing (ADP) system user and, in some instances, to grant or deny access to private

or shared data. This Standard recognises that passwords are not the only method of personal authentication, nor does it endorse the use of passwords as the best method; however, it recognises that passwords are widely used in computer system and networks for these purposes. In these systems and networks, compliance with this Standard will ensure that passwords are used in accordance with accepted practices. This Standard specifies basic security criteria for two different uses of passwords in an ADP system, first, personal identity authentication and second, data access authorisation. A password used for personal identity authentication will be called a personal password; a password used for authorising access will be called an access password. A personal password should not also be used as an access password. This Standard does not require the use of passwords in an ADP system for either purpose, but establishes the basic criteria for the design, implementation and use of a password system in those systems where passwords are used.

### 2.3.2 Background

Passwords are the most common method of personal identification used in conjunction with remote terminals to deter unauthorised access to computer systems and networks. The effectiveness of passwords has often been questioned, primarily because they can be easily forgotten or given to another person. However, passwords can provide reasonable deterrence to unauthorised access if properly handled by people authorised to use them and if properly stored and processed in the password verification system. Within its Computer Security and Risk Management Program, the Institute for Computer

Sciences and Technology of the National Bureau of Standards developed this Standard for secure password usage to assure reasonable handling, storage and processing of passwords. This Standard is one in a series of Standards and Guidelines issued by NEBS in the field of Computer Security. Another in this series, Federal Information Processing Standards Publication (FIPS PUB) 48, Guidelines on Evaluation of Techniques for Automated Personal Identification, describes various techniques for verifying identity and provides a set of criteria for the evaluation of automated identification systems embodying these techniques.

Shortly after issuing FIPS PUB 48, NEBS published Special Publication 500-9, The Use of Passwords for Controlled Access to Computer Resources. This publication considered the generation of passwords and their effective application to the problem of controlling access to computer resources. Following analysis and use of this document, a project was initiated to establish a fundamental performance standard for the use of passwords and a guideline on how to use this Standard to achieve the degree of protection that passwords were intended to provide.

The Password Usage Standard was developed within the Computer Security and Risk Management Program of the Institute for Computer Sciences and Technology with considerable assistance from representatives of Federal organisations and private industry. In 1980, NEBS developed and distributed a draft Password Usage Standard to government and industry representatives for comments and then held a workshop to discuss the benefits and impact of the draft Standard. The draft Standard identified 10

factors to be considered in the implementation of password systems and quantified security criteria in a hierarchical manner for each of the 10 factors. It also proposed five levels of security and specified minimum criteria for each level. The workshop participants felt that the 10 factors were useful in structuring the design of password systems, but that the proposed five levels were unworkable as a basis of a password Standard. As a result of the workshop recommendations, the Standard was revised to specify minimum criteria for the factors of a password system. An Appendix was drafted which provided guidelines for achieving higher levels of security. This revised Standard and the draft guidelines were published for public comment and for agency comment in July, 1981. The received comments were used in revising the proposed Standard and draft guidelines in preparing the published Standard and guidelines.

### 2.3.3 Factors

Ten factors of an automated password system are specified in the Standard. These factors constitute the fundamental elements that must be considered, specified and controlled when designing and operating a password system. The rationale for the factors and for the minimum acceptable criteria for the factors specified in the Standard are provided in the following discussion. Guidance on how to meet the minimum criteria and reasons for exceeding the minimum criteria are also provided.

**A) Composition**

A password is a sequence of characters obtained by a selection or generation process from a set of acceptable passwords. A good password system has a very large set of acceptable passwords in order to prevent an unauthorised person (or intruder) from determining a valid password in some way other than learning it from an authorised person (i.e., owner). The set of acceptable passwords should be large enough to assure protection against searching and testing threats to the password system (and hence the data or resources that it protects) commensurate with the value of the data or resources that are being protected. The set of acceptable passwords must be such that it can be specified easily, that acceptable passwords can be generated or selected easily, that a valid password can be remembered, can be stored reasonably, and can be entered easily. Composition is defined as the set of characters, which may comprise a valid password.

The composition of a password depends in part on the device from which the password is going to be entered. It also depends on how and where the password is going to be stored and how the stored password will be compared with the entered password. Federal Information Processing Standards Publication 1-2 (FIPS PUB 1-2) defines several proper subsets of this set to be used for special applications. The 95- character graphics subset specified in FIPS PUB 1-2 is the set from which the System Manager and Security Officer should select the acceptable composition for a particular system. While backspaces can be used effectively to mask printed passwords, several comments on the

draft guidelines described the special use of backspace in many computer systems and recommended that it not be allowed.

**B) Length**

Length is closely associated with composition in assessing the potential security of a password system against an intruder willing to try exhaustively all possible passwords. The length of a password provides bounds on the potential security of a system. A length of exactly 1 reduces the potential number of valid passwords to the number of characters in the acceptable composition set. A length of 2 squares this number; a length of 3 cubes this number; a composition of 10 and a length of exactly 4 provides for 10- (read 10 raised to the fourth power) or 10,000 possible passwords. PIN are typically four digits because of low security requirements, for ease of remembering by a large customer base and for speed and accuracy of entry. A PIN verification system generally prevents a person from quickly trying all 10,000 possible PIN for a particular valid financial account in order to find the valid PIN. If the trial and error process can be automated, even on a small home computer, the valid PIN can be found in a few minutes. Having a length range of 4-6 increases the possible number of PIN to 1,110,000.

If all other factors are temporarily ignored, the security provided by a password is directly proportional to the allowed length of the password. In other words, longer passwords are more secure. However, other factors cannot be ignored in practical password systems. Long passwords take longer to enter, have more chance of error when

being entered, and are generally more difficult to remember (the latter may not be true unless the password consists of random characters). Sixteen random hexadecimal characters are very difficult to remember and very difficult to enter quickly and accurately. For this reason, DES keys are usually not personal passwords and vice versa. However, long passphrases can be transformed to virtual passwords of exactly 64 bits (or 56 bits with the other 8 bits recomputed to be parity bits). Long passphrases can be easy to remember but still take longer to enter. The length range should include a number of lengths, probably from 5-8 characters, and the composition should be a large set so that a high level of security can be provided easily.

A passphrase is an understandable sequence of words (sentence, sentence segment, phrase) that can be transformed and stored as 64 bits, and which is used as a password. A passphrase is generally easy to remember by the owner of the passphrase, and hence is allowed on some systems because of this characteristic. Since the number of distinct possibilities of understandable passphrases is considerably smaller than for a random sequence of characters of the same length, a longer passphrase is preferable to a shorter one.

A passphrase may be used that is equivalent to a password as specified in the Standard. A passphrase may be transformed into a virtual password by using a transformation such as a hashing function or a cryptographic function. These functions should compute a value using the entire passphrase as input such that any change in the passphrase should result in a different computed value (within some probability). The

value that is computed is the virtual password and must be 64 bits as specified in the Standard. This allows all password systems to allocate a maximum of 64 bits for storing each password, and therefore allows up to 264 possible passwords (many thousands of years of security against exhaustive searching attacks).

**C) Lifetime**

The security provided by a password depends on its composition, its length, and its protection from disclosure and substitution. The risk associated with an undetected compromise of a password can be minimized by frequent change. If a password has been compromised in some way and if a new password is created that is totally independent of the old password, then the continued risk associated with the old password is reduced to zero. Passwords thus should be changed on a periodic basis and must be changed whenever their compromise is suspected or confirmed.

The useful lifetime of a password depends on several variables, including:

- The cost of replacing a password;
- The risk associated with compromise;
- The risk associated with distribution;
- The probability of "guessing" a password;
- The number of times the password has been used;
- The work of finding a password using exhaustive trial and error methods.

Password systems should have the capability of replacing the password quickly, initiated either by the user or the Security Officer. Passwords should be changed voluntarily by the owner whenever compromise is suspected and should be changed periodically with a maximum interval selected by the Security Officer. The interval may be a period of time or depend on a number of uses. The password system itself should have automated features, which enforce the change schedule and all the security criteria for the installation. The system should check that the new password is not the same as the previous password. Very sensitive applications may require that a new password not be the same as any of the previous two, three, ..., N passwords. Such a system requires storage for N passwords for each user. It should not be a requirement of a system that the password for each user be unique. Having a new password rejected for this reason confirms that another user has the password.

**D) Source**

Either the owner or the password generator should select passwords at random from the acceptable set of passwords. However, this guidance may not be possible in all cases and may not be desirable in some cases. The Security Officer often selects a password for a new user of a system. This can be used for the first access to the system. The system may then require that the user replace this password which the Security Officer may know with a password that only the user knows. Passwords that are created or selected by a user should be checked by the automated password system as meeting all of the criteria of the password system. Passwords that do not meet all the criteria should

be rejected by the automated password system. A record that an attempt to select an unacceptable password may be made by some automated systems but is not required by the Standard.

If passwords are generated by the system, the method of generation should not be predictable. Commonly used random number generators that are available in computer systems for statistical purposes should be avoided because the sequences of random numbers that they generate are predictable. The DES algorithm, together with a non-deterministic parameter such as the least significant bits of a high-resolution computer system clock may be used. The results of a random generator are then combined with password selection rules to obtain a password that meets mandatory and desirable criteria.

**E) Ownership**

A personal password should be individually owned rather than owned in common by a group of individuals in order to provide individual accountability within a computer system. This is desirable even though a group of people all has common access privileges to the same resources or data. Individual ownership of personal passwords is required because:

- It can establish individual accountability for the determination of who accessed what resources and for what purposes.
- It can establish illicit use of a password or loss of a password.

- It can be used for an audit trail of the activities of a user.
- It avoids the need to change the password of an entire group when a single member of the group leaves or loses authorisation privileges.

**F) Distribution**

A password must be transported from the owner to the authentication system if selected by a user, from the authentication system to the owner if generated by the password system or from the Security Officer to both the owner and the authentication system if generated by the Security Officer. The initial password is often distributed in a different manner than subsequent replacement passwords. The initial password is generally created and issued directly, either orally or in writing, Curing the meeting at which a user is initially authorised use of the computer system or access to a set of data. This may be a one- time password, which must be changed after the initial access request is granted. Changing of a password by a user generally requires that the user supply the old password and then the replacement password. The replacement is checked for meeting the security requirements of the system, checked that it is different than the old password, and then entered into the storage location of the old password. An audit record should be made of the replacement, containing the date and time of the change, but not the new password. Forgotten passwords should be replaced and a new password issued in a manner similar to, if not identical with, issuance of the initial password. Passwords that are distributed in writing should be contained in a sealed envelope marked "To be opened by addressee only". Instructions to the user should be to:

- Destroy the written password after memorising it; or
- Return the written password to the Security Officer after signing the receipt for the password and after sealing it in the return mailer.
- Use the password as soon as possible and, if the user can change the password, change the password.

Some systems distribute passwords in a sealed mailer that has been printed by a computer. The mailer is designed so that it cannot be resealed once it is open. The password is printed only on the inside of the mailer on the second page using carbon paper attached to the back of the mailer's front page. The instructions say to remove the front of the mailer, which shows the name of, 'the intended recipient, to destroy the front and save the password (in a protected place readily accessible only to the intended recipient). The part of the mailer that has the password has no other identification, which would associate the password with either the system or the owner. Thus, anyone finding a lost password would usually not be able to use it. While not as desirable as memorising the password and destroying the distribution medium, this system is useful when passwords are not routinely used and would be written in a location which-is more easily associated with the owner.

When distributed by a secure mailer, a receipt for the password may be validated by positive response or on an exception basis. When password distribution is done on an unscheduled basis, a positive response is required. When passwords are distributed regularly, the user should be expecting a new password and should report any failure to

obtain a new password. In either case, a record must be kept of the fact that a new password was issued.

There may be a transition period in which it is uncertain if the old password is valid or if the new password is valid. Some systems may allow either password to be valid during the transition period. This means that both passwords must be stored and compared with an entered password. Some systems may have no transition period (e.g., a password becomes valid at 8:06 P.M. exactly) and record attempts at using the old password in an audit file. A report of such attempts should be sent securely to the password owner as notification that usage of an old password was attempted. The owner can verify that the use was an accidental rather than an unauthorised use of an old password by an intruder.

**G) Storage**

Passwords should be stored in the authentication system in a manner, which minimises their exposure to disclosure or unauthorised replacement. Several methods have been used to protect passwords in storage. Most systems have a password file that can be legitimately read only by the "LOGON" program. The file is protected by a file access mechanism which checks a protection bit in a file access table. Only the privileged LOGON program has access to read the file and only (the password program has access to write the file. Some systems separate the password file from the authorised user file. An index file is used to provide the correspondence between the user and the user's password. Some systems encrypt the passwords, either reversibly (two-way) or

irreversibly (one-way) using a Data Encrypting Key (DEK) or the password itself as a key. Of course, any key (e.g., a Data Encrypting Key) retained in storage would also need protection by encryption using a Key Encrypting Key (KEK). The type of protection provided to the passwords should be commensurate with the protection desired for the system or data and hence a protection system should be used to provide the desired protection.

One-way encryption of passwords is allowed in the Standard when encryption is used for stored password protection. One-way encryption systems transform the password in such a way that the original password can not be recovered. This protects the original password from everyone, including the Security Officer and the systems programmers. When a user is logging onto such a system, the password that is entered by the user is one-way encrypted and compared in encrypted form with the stored encrypted password. The same encryption method and key must be used to encrypt the valid password before storage and to encrypt the entered password before comparison.

Two-way encryption of passwords is also allowed in the Standard. Given the correct key, the original password may be determined from the encrypted password. A user-entered password may be compared with the decrypted stored password (which was encrypted), or the user's password may be encrypted and compared with the stored password as is done with one way encrypted passwords.

## H) Entry

Entry of a password into an automated authentication system in a secure manner is often a difficult task. An observer often is able to detect part or all of a password while the user is entering the password. Typing keyboards are the typical entry devices. A user that is not a trained typist often enters the password with one finger. A long, random password that is difficult to enter may be more vulnerable to observation than an easily entered password. The Standard specifies that a password shall be entered by a user in such a manner that the password will not be revealed to anyone observing the entry process. The following discussion provides some techniques, which the user may find useful in achieving this goal and which the computer systems operation staff may find useful in assisting the user.

The computer terminal, keyboard, push buttons, or password entry device should provide a means for minimizing the exposure of the password during entry. The password should not be printed on the terminal during the entry process. If the keyboard and the terminal display or printer are directly coupled, then the password should be masked by obliterating (understriking) the space where the password is going to be printed. Overstriking the area after password entry may mask the password further. Computer generated masks used during password entry to disguise the entered password should not always be the same. In any case no printed or displayed copy of the password should exist after password entry.

CRT terminals that use half-duplex communications may present a problem because the password overwrites the understriking and remains visible on the display. The password entry program should immediately clear the display after password entry in such systems. Users should be instructed to manually clear the display following password entry if the password entry program cannot clear the screen.

When submitted as a part of a remote entry batch processing request, the password should be added to the request at the last possible moment and physically protected. Batch processing requests submitted in punched cards should have the password card added by the user just prior to submission. The computer operations staff should maintain the card decks in a protected area and should remove and destroy the password card after the deck has been read by the system. The password should never be printed on any output media. One-time passwords that are distributed to the owner in the form of a password list and sequentially used for sequential batch processing requests may be used. The Standard requires that the owner physically protect such lists.

Users should be allowed more than one attempt to enter a password correctly in order to allow for inadvertent errors. However, there should be a maximum number of trials allowed for a password to be entered correctly. A maximum of three (3) attempts is considered adequate for typical users of a computer system. The system should also prevent rapid retries when a password is entered incorrectly. Several seconds should elapse before another password is requested. This prevents an automated, high speed, trial-and-error attack on the password system. A security record should be maintained of

the fact that incorrect passwords were entered but the incorrect password should not be kept in the record. A security alarm should be generated if:

1. The maximum number of allowed password retries is exceeded;
2. The maximum number of allowed failed logons from one terminal is exceeded;
3. The maximum number of allowed failed logons for a time period is exceeded.

These parameters must be set according to the sensitivity of the data being protected, the profile of the typical system user and the policy of the organisation. Some organisations will be willing to set the parameters high to prevent customer dissatisfaction while other organisations will set the parameters low to prevent security compromises. Terminals should be disabled and users should be denied service if these parameters are exceeded. The Security Officer should be the only one who can enable the terminal and restore the service of the user following these events.

The system should inform the user, following a successful LOGON procedure, of the last successful access by the user and of any unsuccessful intervening access attempts. This will aid in uncovering any unauthorised accesses or attempted accesses, which may have occurred between successful accesses. The user can do several actions to prevent an observer from learning the password by watching the password entry process. First, entry of the password can be practised so that it can be quickly entered using several fingers. Second, the body can be used to prevent the observer from seeing the keys being pressed

during password entry. Third, the user can request that a guest not watch the password entry process. Fourth, the user can perform the password entry prior to demonstrating use of the system.

**I) Transmission**

Passwords are typically used to authenticate the identity of a user attempting to gain access to a shared computer system or network from a terminal. In order to be authenticated, the password is typically transmitted from the terminal to the computer via the communication line between the terminal and the computer. Unless the communication line is physically protected or encrypted, the password is vulnerable to disclosure. Most communication lines between terminals and computers are not afforded this protection at present. Therefore, users should be aware that their passwords could very easily be disclosed via passive wiretapping.

Computer systems can also be easily spoofed. This can occur if an intruder has inserted an active wiretap between a terminal and the computer. An active wiretap can be built today for several hundred dollars by a home computer hobbyist. The wiretap can be built into a briefcase and consists of a hobby computer with a receive/transmit communication chip that receives data from the terminal and computer and then retransmits data to the computer and terminal, having scanned and modified the data. The active wiretap can replace one user's password with another user's password, even if the passwords are encrypted at the terminal. Spoofing occurs when the system is fooled into "believing" one user is at the terminal when another user is actually there. Reverse

spoofing occurs when a user is fooled into believing that communication with the intended computer when another computer is there. In the latter case, an authorized user can be spoofed into providing the valid user's password by simulating the "LOGON" request of the intended computer. After the password is obtained, the intruder that is controlling the spoofing computer informs the user that the requested service is temporarily unavailable. During this exchange the intruder has obtained a valid password without the user's knowledge.

These threats can be prevented by one of two encryption methods. First, the communication line between the terminal and the computer can be protected by encryption devices, which use a secret key (e.g., a Data Encrypting Key) for encrypting all communication between the terminal and the computer. Transmitted passwords are thus protected from disclosure. In addition each transmission can be numbered so that a previous transmission cannot replace a later transmission (i.e., a previously used valid password cannot be saved and used to replace an invalid password, even if both are encrypted). Passwords are thus protected to the same degree as the data as specified in the Standard.

Alternatively, the password can be used as the encryption key or as part of the encryption key. Suppose a user enters a password to be used as an encryption key at the terminal (i.e., never transmitted to the computer) and the user's password is retrieved from the computer's memory and used as the encryption key at the computer (i.e., never transmitted to the terminal). Then the terminal and the computer are mutually

authenticated if normal communication can occur using the encryption and decryption processes at the terminal and computer, both using the password as the key (or a part of the key). This alternative is also allowed in the Standard.

In order to prevent compromise of the level of security provided by the cryptographic mechanism, the Standard specifies that personal passwords that are used as keys as described above are selected at random from the set of all possible encryption keys used by the cryptographic process. It also specifies those passwords that are used as Data Encrypting Keys should not also be used as Key Encrypting Keys, and vice versa. This is to minimise any possibility of attempting to recover the key (and hence the password) through cryptanalytic techniques.

**J) Authentication Period**

Interactive "sessions" between a user and a computer via a remote terminal often last several hours. While security policy should state that a terminal that is "logged onto" a computer should never be left unattended by the user that is "logged onto" the computer, in practice this often occurs. Many systems have a feature, which automatically logs a user off the system if the terminal has been inactive for some period of time. This is to prevent someone who encounters an unattended terminal from using it. Some access control systems require that a user be reauthenticated on a periodic basis in addition to the initial authentication process. These systems often antagonise the user if the authentication frequency is set too high. The message that the authentication process

must be performed again often comes in the middle of the work that a user is performing. If this work happens to be a large printout of final text of a paper to be published, the user is rightfully upset. For this reason the Standard did not specify a minimum reauthentication period. Reauthentication should only be required to satisfy high security requirements, and then only requested if the terminal has been inactive for a period of time. This should prevent the authentication process from occurring in the middle of some important work.

### 2.3.4 Password System for Low Protection Requirements

A hypothetical password system might have the following parameters for the 10 factors that will both satisfy the Standard and satisfy requirements for protection, which are considered to be minimal. The example is similar to that found in much retail, customer initiated financial transaction systems in which the maximum liability of the customer is $50 and the number of transactions allowed per day limits the maximum liability of the bank. This example is also typical of many government-owned, government-leased computer systems in which no sensitive applications are performed. Small scientific systems, special purpose systems and systems not making critical automated decisions may fall in this category. Systems that have limited financial liability and those that require only accountability and control of computer usage and costs may also be considered in this category:

1. Length Range: 4-6
2. Composition: Digits (0-9)

3. Lifetime: 1 year
4. Source: User
5. Ownership: Individual (personal password); group (access passwords)
6. Distribution: Unmarked envelope in U.S. Mail
7. Storage: Central computer on-line storage as plain text
8. Entry: Non-printing "PIN-PAD"
9. Transmission: Plain text
10. Authentication Period: Each transaction

### 2.3.5 Password System for Medium Protection Requirements

Government systems which process limited "sensitive" applications may fall in this category. These are applications, which process data leading to or directly related to monetary payments or process data subject to the Privacy Act of 1974. Agency management may determine that additional applications should be designated as sensitive. Computer systems that are subject to fraud, theft, erroneous payments or other loss of sensitive information may also fall into this category. Government systems which make payments (e.g., Social Security, Treasury), keep inventories (e.g., Armed Forces), and process personal information (e.g., Internal Revenue, Service, Department of Education) would be examples of systems which would have requirements of this nature and probably would be satisfied by this type of password system.

1. Length Range: 4-8
2. Composition: U.C. Letters (A-Z), L.C. Letters (a-z), and digits (0-9)

3. Lifetime: 6 months
4. Source: System generated and user selected
5. Ownership: Individual
6. Distribution: Terminal and special mailer
7. Storage: Encrypted passwords
8. Entry: Non-printing keyboard and masked-printing keyboard
9. Transmission: Clear text
10. Authentication Period: Login and after 10 minutes of terminal inactivity.

### 2.3.6 Password System for High Protection Requirements

Computer systems which process information of a sensitive nature and which rely on passwords to provide personal identification may have high protection requirements that could be satisfied by a password system for personal identification having these characteristics.

Systems having high protection requirement's may include those which have unusually high potential for fraud or theft, have a high economic benefit to a system intruder, and have a substantial impact on safety or the well being of the society. Some computer systems of the Department of Defence or the Federal Reserve Communication System may fall into this category. Systems having very high security requirements may require methods of personal identification which are based on physical characteristics of a person (signature, voice, fingerprint) or on a combination of something unique that the

person has (e.g., badge, ID card) and something unique that the person knows (i.e., a password). A risk analysis should be performed for each government owned or leased computer system to determine its security requirements and then a personal identification system should be selected which best satisfies these requirements.

1. Length Range: 6-8
2. Composition: Full 95 character set
3. Lifetime: One month
4. Source: Automated password generator within the authentication system
5. Ownership: Individual
6. Distribution: Registered mail, receipt required; personal delivery, affidavit required
7. Storage: Encrypted passwords
8. Entry: Non-printing keyboards
9. Transmission: Encrypted communication with message numbering
10. Authentication Period: Login and after 5 minutes of terminal inactivity.

## 2.4 An Enterprise Networking Model

It will help explain how the new technology relates to existing installed technology. The model consists of a number of key components:

### 2.4.1 Telecommunications Facilities

These are various telecommunications services that literally connect the organization to the outside world. Chapter 2 takes a close look at each of these different types of networks, examines their characteristics, and gives hints as to how VPN technology can or will impact their deployment and use. The enterprise telecommunications facilities are "ground zero" for VPN technology. In fact, one of the chief goals of VPN is to move one or more of these telecommunications facilities onto the Internet.

### 2.4.2 Corporate Internet Connection

It's a well-known fact that corporate Internet connectivity is growing by leaps and bounds. Organizations access the Internet for a variety of purposes including the ever-ubiquitous electronic mail (Email), online re-search, and increasingly, electronic commerce (E-commerce), which is an application used to facilitate actual business transactions over the Internet.

### 2.4.3 The Corporate Data Center

The corporate data center contains the "big iron" computing and data servers for the enterprise. In many large organizations the mainframe runs the company's "mission critical," or most important, applications. It is the growth of mainframe usage that, for the

most part, drove the growth of SNA. Today, the SNA network can be found in the telecommunications facility, along with the many other types of internal networks in use. Other mission critical applications run on Unix-based servers or minicomputer platforms that, by the way, are also considered legacy (especially by those seeking to replace them!). These applications include corporate internal E-mail, file and print services, and custodial applications such as Enterprise Resource Planning (ERP), general ledger, payroll, and Manufacturing Resource Planning (MRP). Many of these applications entail network communication to other corporate offices, branches, or warehouses. As we'll see later, one of the main goals of a VPN is to provide secure and cost-effective access to these resources.

### 2.4.4 The Corporate Internet Presence

In addition to simply being connected to the Internet, many corporations are establishing their own presence on the Web. This presence typically includes corporate Web servers and file servers. These may have links to E-commerce systems to enable the enterprise to conduct business on the Web. The design and implementation of many of these systems may be outsource and/or packaged as part of more comprehensive VPN business deal. We will also examine how some E-commerce and Web technology can be seen as a competitor to VPN.

### 2.4.5 The Internet Firewall

Central to connection of the Internet is the ability to let the good and useful traffic in and keep the bad traffic and users out. This is not a trivial task. A whole gamut of devices known as Internet firewalls have been developed to do this. Some firewalls even include VPN services. Generally speaking, VPN will need to coexist and cooperate with a variety of Internet firewalls. Different VPN approaches will have different security issues to deal with. Many of these will be discussed in later chapters.

### 2.4.6 The Enterprise Management Center

In addition to network management consoles that host network performance, fault, topology, and configuration tools, the network management center hosts a number of other infrastructure servers. These include servers for naming, addressing, authorization, and a quickly developing area called "policy." Increasingly, this information is structured to mirror the organization of the enterprise and stored in directory servers based on the Lightweight Directory Access Protocol (LDAP). Finally, VPN often require working with service providers. Monitoring service provider performance and the performance of the VPN, itself, introduces much new management challenges.

### 2.4.7 Campus Infrastructure and Wiring

All of the hubs, switches, routers, and wiring systems that connect all of the enterprise computing elements together represent the infrastructure. These network elements do not cross the public domain and only reach limited distances. They use very high-speed LAN technology that is inherently multiprotocol. Some key systems to consider are:

- The data center system, containing the highest speed switched connections, ties all of the servers of the data center together and connects it into campus and riser systems.
- The campus systems connect various buildings on a single campus together; communication typically stays on dedicated transmission systems over private property.
- The building riser system connects the various floors or departments into the infrastructure. These too, are often high-speed fiber optics-based switched services.
- Horizontal wiring connects the various desktop and end-user computing elements to the building riser. Often this is slower shared copper media, although this, too, is changing to higher speed switched infrastructure as more multimedia applications and powerful desktop appliances appear.

### 2.4.8 Departmental Servers

These include file and print servers used by a small group local to a floor or department. Often, these servers implement legacy protocols such as AppleTalk, IPX, or NetBEUI.

### 2.4.9 End Users

Finally, there are the end users themselves. In many companies it seems that the WinTel (MS-Windows on Intel-based PCs) architecture has won the heart of corporate Information Systems (IS). There are, however, other platforms to consider:
UNIX desktops (especially in engineering environments), Macintosh, and IBM's OS/2, to name a few. It is the users who will gain the most from VPN technology, both from at the desktop and while away from the office.

## 2.5 Application Platform

Unix platform, Linux and Windows NT platform are three of the most famous platforms for developing web client server application. Each has its own weaknesses and strong points.

## 2.5.1 UNIX

Unix is an OS developed at AT&T Bell Laboratories. It is a powerful and mature OS and network based platform.

- UNIX is an increasingly popular operating system. Traditionally used on minicomputers and workstations in the academic community, UNIX is now available on personal computers and the business community has started to choose UNIX for its openness.
- UNIX, like other operating systems, is a layer between the hardware and the applications that run on the computer. It has functions that manage the hardware and functions that manage the executing of applications
- UNIX includes the traditional operating system components. In addition, a standard UNIX system includes a set of libraries and a set of applications. It includes the file system and process control and a set of libraries.
- One of the greatest strength of UNIX is the consistent way in which it treats files. It is very easy for the users to work with files because users don't need to learn special commands for every new task.
- Besides UNIX is not known only for its longevity and versatility as an operating system, but also for the variety and number of utility programs that called tool.

The problem with Unix is that it is too expensive to use. It need very powerful workstations and therefore not cost effective.

### 2.5.2 Linux

Linux is another version of Unix based OS. It is as stable as Unix and best of all it is free. The problem with Linux is that it is developed by people world wide, therefore lack proper organized support.

### 2.5.3 Microsoft Windows NT 4.0

Microsoft Windows NT Server 4.0 is a network foundation for the future, designed to meet the most demanding requirements of today's business computing world. Windows NT Server is also the easiest network operating system to set-up, manage and use. These robust, multipurpose network operating systems offers dependable file and print services, while providing the architecture to run powerful client/server applications. With built-in support for communications and Internet services, Windows NT Server is the only network operating system that includes Internet and Intranet capabilities. The new features built into Windows NT Server offer a better communication by providing more choices for accessing information - especially through a wide range of built-in Internet tools. New features also provide users with easier, lower cost networking and improved performance. Microsoft Windows NT is one of the powerful operating systems for business computing. It integrates a variety of network services that is needed business environment. It is most suitable to be used in LAN environment.

## 2.5.4 The advantages of Windows NT 4.0 over other application platforms

- The Microsoft Windows 95 operating system user interface has been integrated into Microsoft Windows NT 4.0, making the server interface easier to use and consistent with Windows 95 and Windows NT Workstation 4.0. This interface is easy to use and making the task of administrate Windows NT an easy task. It combines the ease-of-use of Windows 95 with the power and reliability of Window NT. NT is also a powerful OS that reliable, secure, multithreaded, symmetric processing, support client/server system.
- Windows NT has been designed in a modular form; the operating system as a whole is made up of separate software components, with each component managing a separate function of the operating system. This modular configuration ensures that future upgrades and extensions to the operating system can be made without having to change the entire operating system. The separate modules of the operating system do not share any program code, and all communication between the modules is done via system calls.
- Window NT uses New Technology File System (NTFS). NTFS is the new technology file system that was developed specifically use with Windows NT and operates in the same way as a relational database. If the system should fail for any reason, the operating system can use the transaction logs to undo or redo any incomplete transactions. By doing this, the file system is left in a known state. With NTFS, local security can be set up at the volume, folder, and file level. Access to folder, volume and file can be logged. This auditing can be used to either log possible security

breaches or to provide an audit trail on which operations have been performed on the file system. NTFS uses a recovery technique known as cluster remapping to recover from disk errors caused by bad sectors. This technique ensures data still can be access when bad sectors occur.

- Remote Access Service (RAS). Windows NT comes with RAS. To gain transparent access to a remote Windows NT network, Microsoft clients can use the Remote Access Service.
- Windows NT allows Object Linking and Embedding (OLE). OLE allows Windows NT to combine information from several applications into one compound document using the special object linking and embedding capabilities of windows based application.
- Windows NT also capable of supporting COM and DECOM.
- Windows NT has built in server and browser.
- Windows NT provides more reliability compare with Windows 95. It uses separate memory spaces for different 16 bit applications. If one application fails, it won't affect other application.
- There is an extensive security support in Windows NT. Windows NT can control the access control of user in accessing certain file or application. This can use for implemented the access control for the Virtual Private Network Web System. Besides, Windows NT supports a wide range of networks protocol and Remote Access Protocol. This makes it easy for us to develop the distributed application.
- Windows NT is a complete platform available for building and hosting web-based application. It is the best platform to publish and share information securely over

corporate Intranet and Internet. It is so reliable that when an application have problem it doesn't crash the whole program.

- Window NT also enables the capabilities of integrating applications on a single computer or even across multiple computers.

### 2.5.5 Comparison between Window NT 4.0 and UNIX

- UNIX is hard to install compared to Window NT 4.0. Sometimes it take two weeks to configure a UNIX machine but only one day to set up a Windows NT box.
- Windows NT 4.0 supports the Microsoft Back Office Product but UNIX doesn't.
- To use a UNIX OS, a lot of command needs to be entered. Compared to Windows NT, it provides user-friendly interface that eases the job of the user.

## 2.6 Database Management System

### 2.6.1 DBMS

- Provide a shared data storage mechanism for users and programs
- The essential task of a DBMS are to provide a means of defining data to the system, storing data on physical devices, and allowing users to access and change the stored data
- DBMS also provide security from unauthorised access, recovery in the event of a system failure, concurrency so that multiple users can access the database at the same

time, and integrity checking so that data in different parts of the database remains logically consistent

Database technology is used in a variety of applications. Some serve only a single user on a single computer while others are for multi-user. There are variety types of database like Microsoft SQL Server 7.0, Oracle 8i, Sybase, Informix and also Microsoft Access.

In order to choose a reliable database, the database must be able to ensure the safety and security of the data. The database is at the core of all mission-critical business applications. Choosing the wrong database can have drastic downstream results. The investment in software, implementation, and development of a database system is substantial; it needs to be able to evolve with the changing requirements of a growing company.

### 2.6.2 Microsoft Access

Access is a database management system (DBMS). Like other products in this category, Access stores and retrieves data, presents information, and automates repetitive task (such as maintaining accounts, performing inventory control, and scheduling).

Access is also a powerful Windows application-probably the best end user/developer product ever written. Microsoft Access brings the productivity of database management

to the usability and consistency of Microsoft Windows. Because both Windows and Access are from Microsoft, the two products work very well together. Access runs on The Windows 95, 98 or NT platforms, so all the advantages of Windows are available in Access.

The Objects give the end-user the capability of creating tables, queries, forms and reports easily. You can perform simple processing by using Expressions, also known as functions, to validate data, enforce a business rule. Macros allow for automation without programming, whereas VBA (Visual Basic for Applications) code lets the user program complex processes. Finally by using Windows API (Application Programming Interface) calls to functions or DLLs (Dynamic Link Libraries) written in other language such as C, Java, or Visual Basic, a programmer can write interfaces to other programs and data sources.

### 2.6.3 Microsoft SQL Server 7.0

Microsoft SQL Server 7.0 combines the best of traditional mainframe computing centralised security, data integrity, and control with the best of today's PC ease of use, rich user interfaces, and a variety of off-the-shelf productivity tools. It makes it possible for multiple front ends to share information, enabling the developer to choose the most appropriate tool for the job. SQL server makes efficient use of network, because database queries are processed at a centralised server, network traffic is reduced. SQL server can comfortably handle databases of 200GB of data today, and that number is expected to

grow to between 500GB and 1 terabyte with upcoming versions of SQL server. Microsoft SQL Server incorporates a world-class feature set for distributed client/server computing. Benefits of using SQL Server are as following key areas:

- Reliable distributed data and transactions.
- Centralised control of distributed servers.
- Very high performance and scalability.
- Support for very large databases.
- Full programmability and standard support.
- Rich desktop integration
- Open interoperability

Microsoft SQL Server 7.0 provides robust, efficient, and effective solutions for complex business problem in the Windows NT environment. A host of features, such as the powerful SQL language, an intelligent cost-based optimiser, build-in replication, a versatile locking strategy, and a distributed transaction control mechanism, make Microsoft SQL Server 7.0 ideal for all kinds of client/server computing environment. SQL Server also provides tools for database administrators for centralised management of SQL Server.

### 2.6.4 Microsoft's Internet Information Server (IIS) 4.0

Microsoft's Internet Information Server is web server software on Windows NT environment. Active Server Page is build into IIS that is bundled with Windows NT. Web pages that are coded in ASP will send to IIS where the ASP script written in Visual Basic or the asp.dll file will interpret JavaScript. This dynamic link library file is stored in the IIS. The IIS locates the page and sends it back to the browser, which the page source codes, is pure HTML. The asp.dll file in IIS has interpreted all the Visual Basic code. Without Internet Information Server, ASP script cannot be interpreted and the browser will read just the static HTML code.

### 2.6.5 Microsoft Transaction Server (MTS)

Microsoft Transaction server (MTS) is a vital component in IIS where IIS cannot be installed without MTS. MTS is used to manage IIS Web application. Web application can be run as separate packages under MTS. This approach allows the program to isolated and maintains IIS integrity. MTS is managing Internet Server Application Programming Interface (ISAPI) application, it can stop one application and the visitor will simply get an error message. MTS tracks any changes that are being made to a database. All changes tied to a transaction must be fail or pass together, even if the transaction includes several normally viewed as separate transactions.

The best way in handling any database transactions at Web site through ASP is MTS. MTS can be used to manage database transaction. A transaction is an operation that succeeds or fails as whole; even it involves a lot of steps in between.

## 2.7 Microsoft Data Access Technology

Data access technology enable communication and access to its various database. Data assessment is very important for each application. In order to access, retrieve and share information efficiently, Microsoft Data Access Technology provides us with a lot of functionality. It provides us with ease-to-use, programmatic access to all types of data throughout the enterprise. It is easy to integrate information from a variety of sources, which is relational (SQL) and non-relational. These components can be used by the data driven client/server applications deployed over the Web or LAN.

There are many types of Microsoft's Data Access Technologies, which are VB SQL, Open Database Connectivity (ODBC), Data Access Object (DAO), Remote Data Object (RDO), ActiveX Data Object (ADO) and OLE-DB.

### 2.7.1 Open Database Connectivity (ODBC)

Open Database Connectivity is one method that used by Visual Basic to communicate with client/server databases. It is a component of Microsoft's Windows Open System Architecture (WOSA). It provides a set of application program interface

(API) functions, which makes it easier for us to connect to a wide range of database formats, that is it supports SQL. We can also access a number of PC databases using ODBC functions.

Open Database Connectivity (ODBC) technology provides a common interface for accessing heterogeneous SQL databases. ODBC is based on Structured Query Language (SQL) as a standard for accessing data. This interface provides maximum interoperability: a single application can access different SQL Database Management Systems (DBMS) through a common set of code. This enables a developer to build and distribute a client/server application without targeting a specific DBMS. Database drivers are then added to link the application to the user's choice of DBMS. Benefits

- ODBC Applications are not tied to a proprietary vendor API.
- ODBC 3.0 aligns with, and is a superset of, the X/Open and ISO Call Level Interface (CLI) standards.
- Increased performance.

ODBC is based on the X/Open Call-Level Interface and uses SQL. During the run time, ODBC driver will communicate with other drivers and through a standard interface called Service Provider Interface (SPI). It is a network independent technology because it employs replaceable network libraries. Figure 2.1 shows the ODBC architecture.

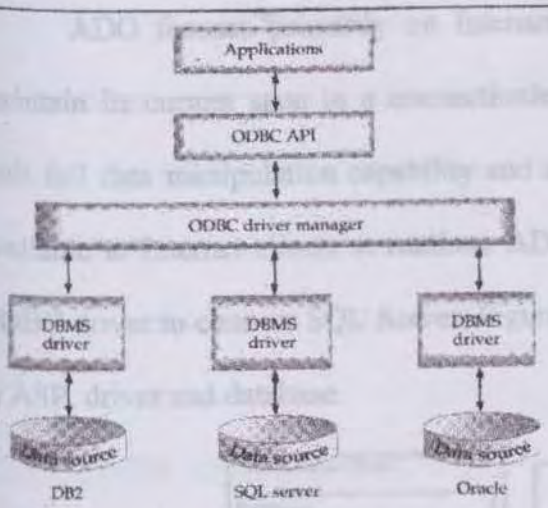


**Figure 2.1: ODBC Architecture: The Application on Top of it**

However, the biggest downside to ODBC is that it must be able to support the capability to translate calls. This means that additional processing overhead can slow the data access a bit.

## 2.7.2 ActiveX Data Objects (ADO)

The ADO is based on an object model that exposes the collections, methods and properties necessary to access and work with the database. This object model is available from ASP code and works conjunction with the OLE-DB layers. It is a new technology for data access based on existing technologies and endowed with increased flexibility. It is an evolution of both DAO and RDO into a single, simplified and extensible interface that will supersede all DB-Library, DAO & RDO functionality.

ADO focuses primarily on Internet deployment because it has the ability to maintain its current state in a connectionless environment. It includes implementation with full data manipulation capability and a downloadable, lightweight implementation available to Internet clients at runtime. ADO base in ASP application works using the ODBC driver to connect SQL Server. Figure 2.2 show the connection between the ADO in ASP, driver and database.

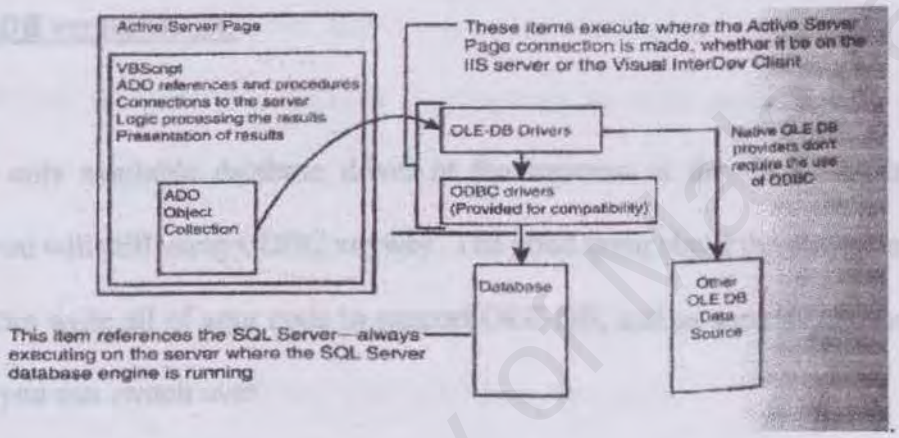


**Figure 2.2: Flow of ADO Connection to Database and Data Sources**

## 2.7.3 OLE-DB

OLE-DB components consist of data providers, which contain and expose data; data consumers, which use data; and service components, which process and transport data (such as query and cursor engines). OLE-DB interfaces are designed to help components integrate smoothly. In addition, OLE-DB includes a bridge to ODBC to enable continued support for the broad range of ODBC relational database drivers available today.

OLE-DB is Microsoft's new low-level database interface that provides access to much different kind of data. It is the extension of application capabilities beyond the limitations of ODBC. It is a COM-based API with features that provide access to both relational (SQL) and non-relational data sources. It provides an environment where database components can be replaceable.

### 2.7.4 OLE-DB versus ODBC

The only available database driver at the moment is the OLE-DB driver for ODBC, so you will still using ODBC anyway. The good thing about the current approach is that you can write all of your code to support OLE-DB, and as soon as the new driver is available you can switch over.

## 2.8 Application Software

### 2.8.1 Microsoft Visual Basic 6.0

In Visual Basic, developers create applications by assembling components. Components can include visual Active-X controls, non-visual Active-X servers and Active Documents. These components are meant to encapsulate some business function, allowing developers to reuse components in many different applications.

Visual Basic is based on a component model rather than an inheritance model. It promotes the encapsulation of business logic and other application functions into reusable

components. Visual Basic also makes the physical deployment of these components simple to ensure that reuse is optimal and load balancing is straightforward.

Instead of stressing inheritance from ancestor objects, Visual Basic encourages developers to create application elements from smaller parts. Visual Basic is built around the premise of two important methods for building components: composition and aggregation. Composition involves the extension of existing components through the inclusion of new methods or properties. Aggregation involves grouping one or more small components together to create a single, new component. Both aggregation and composition are generally regarded as better forms of reused because they do not rely on inherited changes from ancestor components. Visual Basic's code reuse model also forces developers to think of reuse as it correlates to business processes.

### 2.8.2 Java

Java is two things: a programming language and a platform.

#### 2.8.2.1 The Java Programming Language

Java is a high-level programming language that is all of the following:

- Simple
- Architecture-neutral
- Object-oriented
- Portable

- Distributed
- High-performance
- Interpreted
- Multithreaded
- Robust
- Dynamic
- Secure

Java is also unusual in that each Java program is both compiled and interpreted. With a compiler, you translate a Java program into an intermediate language called *Java bytecodes*--the platform-independent codes interpreted by the Java interpreter. With an interpreter, each Java bytecode instruction is parsed and run on the computer. Compilation happens just once; interpretation occurs each time when the program is executed. This figure illustrates how this works.

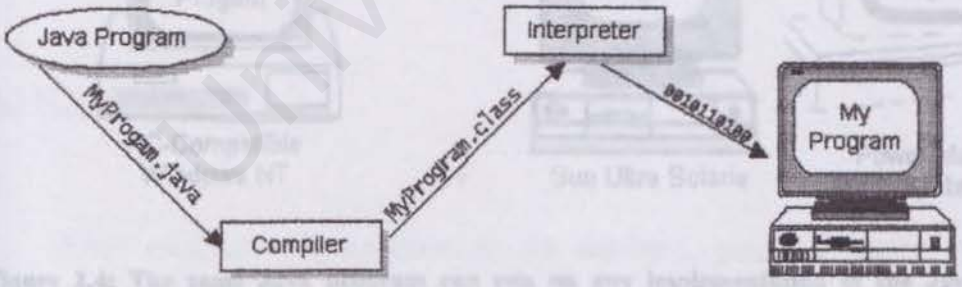


**Figure 2.3: Communication between Java program and computer**

You can think of Java bytecodes as the machine code instructions for the *Java Virtual Machine* (Java VM). Every Java interpreter, whether it's a Java development tool or a Web browser that can run Java applets, is an implementation of the Java VM. The Java VM can also be implemented in hardware.

Java bytecodes help make ***"write once, run anywhere"*** possible. You can compile your Java program into bytecodes on any platform that has a Java compiler. The bytecodes can then be run on any implementation of the Java VM. For example, the same Java program can run on Windows NT, Solaris, and Macintosh.

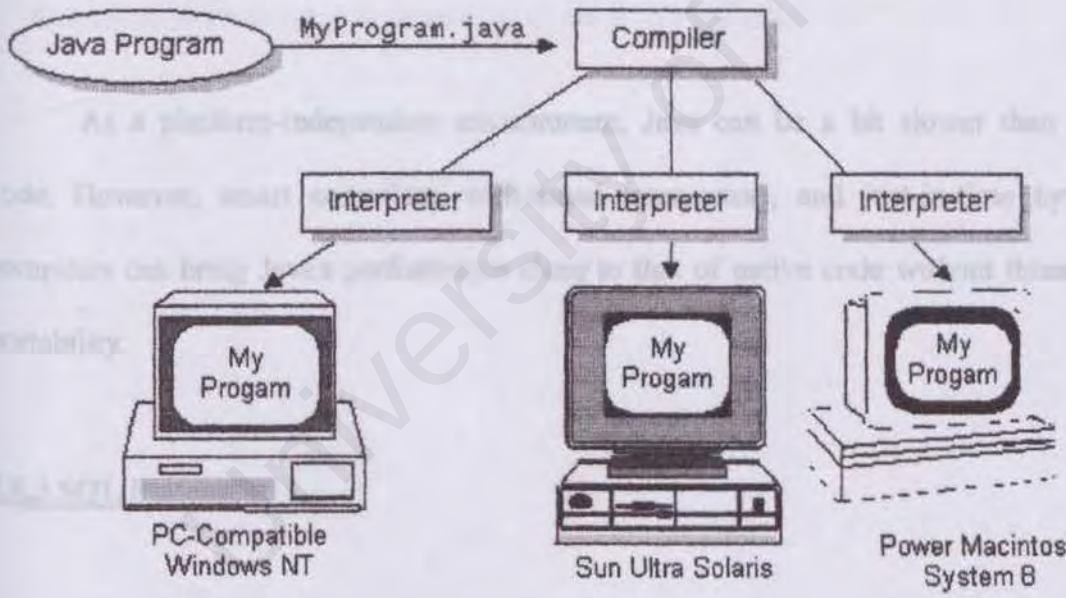


**Figure 2.4: The same Java program can run on any implementation of the Java Virtual Machine.**

### 2.8.2.2 The Java Platform

A platform is the hardware or software environment in which a program runs. The Java platform differs from most other platforms in that it's a software-only platform that runs on top of other, hardware-based platforms. Most other platforms are described as a combination of hardware and operating system.

The Java platform has two components:

- The *Java Virtual Machine* (Java VM)
- The *Java Application Programming Interface* (Java API)

As a platform-independent environment, Java can be a bit slower than native code. However, smart compilers, well-tuned interpreters, and just-in-time bytecode compilers can bring Java's performance close to that of native code without threatening portability.

### 2.8.3 SQL Statement

After establish a connection to the database, you can access it with SQL statements. Most ASP applications that access database use recordsets to retrieve and display records, rather than edit them remotely.

SQL is another popular way to implement queries. It uses a series of keywords and commands to select the rows and columns that should be displayed in the resulting table. SQL is a specialised set of programming commands that enable the developer or end user to do the following kinds of tasks:

- Retrieve data from one or more tables in one or more databases.
- Manipulate data in tables by inserting, deleting or updating records.
- Obtain summary information about the data in tables, such as totals; record counts; and minimum, maximum, and average values.
- Create, modify, or delete tables in a database (Access databases only).
- Create or delete indexes for a table (Access databases only).

# CHAPTER 3 SYSTEM REQUIREMENT AND ANALYSIS

## 3.1 Project Methodology

The system development life cycle used in this project to development the software and this software will be use in the project. This system is powerful and systematically, the project will be executing step by step follow the processing system. From the guideline SDLC (system development life cycle), that is a very complete and suitable method for develop a project from the beginning until the end of the project. After analysis the system and compare with another system, this is the most popular system use in this time. The system use the seven phase for development a project and they are list below:

1. Planning
2. System Analysis
3. System Design
4. Programming
5. Testing and Evaluation
6. Documentation
7. System Implementation

These phases will provide a systematic and consistent approach to work out the new automation system. By considering each phase at a time, these enable us to concentrate on the important issues at any point in time.

The output of each phase are the input at the following step phase or in another word, the next phase is to use the information gathered from the previous step. Therefore, each phase in the project methodology are worked in the sequence order, but it is impossible to clear cut each phase or it may happen in such a situation that two phases are carried at the same time in order to improve the efficiency of overall compatibility.

As a result, we could check the program and estimate the required effort and time frame to complete this project. After the testing and documenting, the final project report will be submitted and that come to the end of the project.

### 3.1.1 Planning

The very first phase in the process involves the recognition and diagnosis of a current system. The definitions for a new system because it can help to develop a new system in the future. During this phase, a good planning step by step need in the new system. The introduction to projects defined what the projects will do it and objective help to know the projects more clearly. In addition, the scope and goals of projects planning can achieve the requirement. Finally, overview for projects can be producing. A project scheduling will be creating to help the project implementation.

Finally, develops a timetable in order to:

a) Examine the current procedures and information flows

b) Pinpoint the problems and difficulties in the existing system

c) Identify what the resources are used

d) Discuss with the management concerning what improvements they needed

### 3.1.2 System Analysis

The second phase of the process involves a detailed look at the current system. This involves conducting an observation, interview and analyzing document, records, and define the data requirements. The focus of analysis is list below:

**a) Review the current system**

This part includes the fact-finding in the current system, some shortcoming and problem occur in the system.

**b) Identify the functional requirements and non-functional requirement**

For non-functional requirement include the user-interface, user-friendliness and the response time in system. The functional requirement includes the database management, the form processing and the available room report.

**c) Analysis of current system**

Defined the processing and the goals for the current system. Data Flow Diagram current system is draw here.

**d) Reference of existing system**

The reference for corresponding system, the system is something like the new system, which is, will be developers.

**e) Resource surveyed and internet definitions**

The resource maybe is using in the new system. The consideration resource for advantage and disadvantage are using. The definitions can know more about the Internet, just like the new system that is on-line system.

**f) Web Application Development Tools**

Defined the all software are most popular use in the web and a consideration about the advantage and disadvantage software. The conclusion for use the type of software will be choice to developer the new system.

### 3.1.3 System Design

The third phase is the job of analyst to conceive the new system and puts it down on paper. He or she has to evaluate software and hardware requirements, writes specifications, design databases, and develops the procedures for querying, revising and outputting information.

The aim of carrying out this phase is to produce a new system which is better that existing in terms of performance, reliability, ease of use and cost saving.

After analyzing the gathered facts, the system is to be designed based on the purpose of saving the problems encountered in the faculty management system as well as achieving the objective which is previously defined.

The primary objective is to define the functional requirements based on user's need and translate it into system with hardware and software constraints. The jobs of this stage are to:

**a) Define the design consideration**

The important design includes the logical directory structure, effectiveness of user interface, size of database, efficiency of process.

**b) Define the process design**

Process design uses the structure chart and module relationship. Finally the data flow diagram for proposed system come out, that the design proposed system more clearly. The detailed description about the relationship in the system can view the entity relationship diagram.

**c) Define the database design**

The data dictionary about the database are describe in the table, it's include the field name, data type, size and the description.

**d) Define the user interface design**

What are the users use the system and the main page will come out to the users.

In this phase, user interfaces are designed.

### 3.1.4 Programming

The output of this phase is to code the programming work units into fully tested, executable modules which prepared for the system tasks in order to:

a) Code the programming work units

e) Perform a detailed code review

f) Prepare test conditions, test data, and expected results

g) Conduct the initial test

The Programming language that is to be used in this project is Visual Basic 6 on personal computer and Microsoft Access for the database.

### 3.1.5 Testing and Evaluation

The objectives of this phase is verify the new system generated in the previous phase contains all the required function and ensure that the functions are performing well and accurately in order to:

a) Create system test model and test condition

b) Examine the expected results from the test models

c) Verify the detailed result

### 3.1.6 Documentation

Documentation should be done throughout the project. The final collection is done at the completion of all the stages in planning, analyzing, designing, programming and testing, All the reports and conclusions which have been previously gathered and rearranged in order to give the client a full report about the whole process throughout the whole system which included:

a) System overview narrative

b) Program descriptions

c) Screen display and sample reports

d) Output prompt and error message explanation

e) File design layout

So far for this proposed, four chapters are write in documentation, that is

**Chapter 1**: Introduction of VPN-Virtual Private Network, objective, scope, goals for projects and the project scheduling.

**Chapter 2**: Literature review such as the system analysis and synthesis project. This chapter include the fact-finding, requirement analysis, analysis of current system, reference to the existing system, list of resources surveyed and the application development tools.

**Chapter 3**, Project Methodology, the project implementation uses the system development life cycles, that are include the seven phases: Planning, System Analysis, System Design, Programming, Testing and Evaluation, Documentation, System Implementation. This chapter also includes the functional and non-functional requirement for the project.

**Chapter 4**, System Design include the process design, module relationship, data flow diagram and entity relationship diagrams for proposed system. Also have the database design and the user interface design.

### 3.1.7 System Implementation

After the new facilities have been acquired, they will be installed and the old system will be removed. This requires planning and adherence to a changeover timetable system. Once installed, the system is fined-tuned, or optimized and the system documentation completed.

The implementation starts with the program testing, periodic reviewed from time to time and modified the system in order to provide an economical, technical and operational feasibility and achieve the objective of the organization.

## 3.2 System Analysis

System analysis is the most important phase in a software life cycle. It is the process of defining a problem, gathering pertinent information, developing alternative solutions and choosing among those solutions. Following are some of the objectives of the analysis:

- To study the problem faced by the user and find out the best solution to reduced it.
- To study how the record is managed and improves it.
- To acquire knowledge on how this system will be developed with the new emerging technology.

## 3.3 Requirements Gathering

A requirement is a feature of the system or a description of something the system is capable of doing in order to fulfil the system's purpose.

The purposes to determine the requirements for a software-based system are :

i) to explain the developers' understanding of how their user wants the system to work.

ii) to tell the designers what functionality and characteristics the resultant system is to have.

iii) to tell the test team what to demonstrate to convince the user that the system being delivered indeed what was ordered.

## 3.4 Characteristics of Requirements

Requirements describe not only the flow of information to and from a system and the transformation of data by the system but also the constraints on the system's performance. In order to ensure the developers and the customers understand and use the requirements properly, it is important that the requirements are clearly defined. Thus, the requirements have the following characteristic: describe something that is needed by the customer, correct, consistent, complete, realistic, verifiable, and traceable.

## 3.5 Requirement Elicitation

Before the requirement is captured, a few techniques have been used to find out what the users really want. The requirement elicitation takes quite a long time. This is due to several techniques needs to be applied in order to get a complete requirement. Following are some the techniques that have been used for this purpose:

- Review the current situation
- Interview
- Internet Research
- Brainstorm

### 3.5.1 Review the Current Situation

Find out the functionality and the problem faced by the current system. We also review whether it is economic to apply the new system and whether there is enough equipment to develop the new system.

### 3.5.2 Interview

Interview activities have been carried out in order to understand the loopholes and weakness of the current system. Through interviews, we can understand better the detail about the whole process of the management. We have found out how the daily record is stored inside the database and how the system functioning. We discovered a lot of problem in handling the current Inventory System, which is not automated.

### 3.5.3 Internet Research

Internet is used as the main resource for referring any ambiguities that arise during the entire development period. Through the Internet, we not only collect some idea from the similar system, we also find out some interesting web design and feedback from the other remote users.

### 3.5.4 Brainstorm

During the requirement elicitation, we meet together with our supervisor and teammates to discuss about the new system. During this stage, we generate as many ideas as possible without any analysis until all the idea have been exhausted. Besides, we study the feasibility of the requirement identified in this stage. After complete the above technique, the requirement is separated into two categories:

- Requirements that absolutely must be met.
  - Only the authenticated user is allowed to use the system. Password and login ID need to be entered to gain access to the system.
  - Some of the control keys need to be disabled in order to protect the system for being hacked.
  - The system should be able to detect the errors and prompt the user.
  - The system should provide the functionality for the user to change.
  - The system should let the user view the important criteria.
- Requirements that are highly desirable but not necessary
  - Tool Tips should have for every control inside the system.
  - Help files should be provided in every window.
  - Error messages with some guidelines should be provided to the users when illegal operations happen.
  - When error occurs, a mail is automatic generated and sent it to the administrator.

## 3.6 Functional Requirement

Functional requirement explains what the system will do, independent from the implementation of the solution. It describes an interaction between the system and its environment to determines functional requirement, a decision has to be made on what states are acceptable for the system to be in.

### 3.6.1 Authentication System

An authentication system is needed for ensuring the integrity of the system. This authentication system is responsible for the security of the system. User is required to enter password and user ID upon logging into this system. This is to prevent unauthorised users from accessing this VPN- Data Center System.

### 3.6.2 Data Management

Data management ability is needed in the VPN-Data Center System. The sub system that provides this functionality will enable user to search or query, and retrieve the data from database for viewing. A network service company will require maintaining a large record of tables like companies, vendors, employees (engineers, clerks, and managers), and branching. For user, this function will enable user to read the product's catalogue before making any purchase. Therefore the Database System must be capable to managing these records and enable data updated automatically in the related tables.

### 3.6.3 Users/Administrators Management

The system must enable authorized user to read, change, edit and update their personal details like name, address, telephone, and fax number.

### 3.6.4 Inventory Control Management

A search capability is also needed to enable the ease of looking for a particular record especially the branch management to control and monitor the status of stocks in various branches. The branch management must be able to let the manager at main store to monitor or checking the inventory in branch. Detail of every records available amount at branch can be listed. The system also must enable the manager at main store to order moving of stock to branch or move stock from branch back to main store.

The system must be able to order stock from various vendors. The system must provide a form to let user enter order details. These details include amount of products, price and to whom the order is forward. The order then must be able to post up at the online server. This stock ordering functions must also handle stock arrival. It is capable of automatically updated the in store inventory when amount of arriving stock is enter into the system. The remaining amount of stock ordered but still haven't arrived will remain in its record.

## 3.7 Database Requirement

A large amount of data is required to store in the database. Therefore a database that capable of handle large amount of data is required. Since replication is required to archive through Internet, a database server that supports stable replication through Internet is required.

Various tables are stored all the data in the system. Relationship between these tables must be designed carefully to ensure integrity of the data. Different tables are needed to replicate through Internet. Therefore, the database must be design and create to ensure the data enter through VPN-Data Center System can update to this database systematically.

## 3.8 User Interface Design Requirement

The user interface design in the VPN-Data Center System would be as user friendly as possible. It must provide a fast learning curve for user of the system. The user interface itself must be self –explainer where user can understand the functions of the system just by trying out the user interface.

The user interface must be divided into main section and navigator bar section. This will enable user to navigate from one function to another easily. The colour scheme used for

the user interface must be carefully selected in order to ensure it is pleasant to behold. The icons used must be able to provide clearly idea to user about the functions that they represent. A general description of the functions in each module must be given in the user interface to ensure user operate the system easily.

## 3.9 Non-Functional Requirement

In order to ensure the quality of system produced, it must conform to certain software quality factors. The proposed solution for the VPN-Data Center System must follow these non-functional requirements:

### 3.9.1 Flexibility

The VPN-Data Center System must be able to incorporate new technologies in the future and in fast changing environment. These technologies include Object Oriented technology and advance security technology.

### 3.9.2 Usability

The system must be user friendly. User must be able to use VPN-Data Center System in the shortest learning curve. The VPN-Data Center System can be customised to meet the need of changing business rule and process. Interfaces must be self-explainer and consistent with other application in the environment.

### 3.9.3 Correctness

The final VPN-Data Center System must meet the objective, specification and requirement of the users stated earlier. The VPN-Web Based Data Center System will be build according to the user requirement and specification.

### 3.9.4 Scalability

The VPN-Data Center System must be capable of migrate or move from machine with different specification, with minimum or no changes to the underlying component. It must be able to meet this requirement as the basic structure of hardware and software environment is changing constantly. This can be archived, as the VPN-Data Center System will be build as a web application.

### 3.9.5 Reusability

This components and different part of this system must be capable to reuse. Components and parts are required to be self-contained in order to archive reusability. This requirement is important as to support future redesign or expansion of current system. Each module of the VPN-Data Center System can be used separately with minimum or little modification.

### 3.9.6 Portability

Portability of VPN-Data Center System will enable the application to work on various platform, hardware and operating system. Components in this system will be designed to ensure migration of component does not or only require minimum modification, recompiling, reconfiguration or redesign.

### 3.9.7 Maintainability

This application is designed so that the effort required to maintain, locate and fix an error in the program is minimum. Adequate comment is required to ensure the application is easy to maintain.

### 3.9.8 Manageability

The VPN-Data Center System should be capable of being managed and operate easily.

### 3.9.9 Users and human factors

System is easy to use. No special skill is required.

### 3.9.10 Performance

The pages can load easily with a minimum delay time and the changes are immediate updated.

## 3.10 Programming Tools

### 3.10.1 Visual Basic

Visual Basic is a programming language for Windows that is often used to organize and present multimedia elements. It is made up of control (objects) that reside on forms (or windows). Visual Basic uses language code syntactically similar to BASICA or GW-BASIC. The program is event driven, that is, code is attached to objects and remain idle until called to respond to user or system-initiated events, such as mouse click or system timeout [Vaughan, 1998.].

Using integrated visual database tools, advanced database applications can be developed to access SQL Server database, Access database or any third-party database by using ODBC, DAO, RDO, or ADO and bind the data to forms and reports which greatly reduce development time [Mckelvy, 1997].

Besides, it provides support for Graphical User Interface (GUI) design, which helps interface designer to enhance screen design. We use controls to create the user interface of an application, including command buttons, option buttons, check boxes, list

boxes, combo boxes, text boxes, scroll bars, frames, files, and directory selection boxes, timers, and menu bars [Mckelvy, 1997].

## 3.11 Database Implementation

### 3.11.1 Microsoft Access 97

It is a relational database management system. It integrates data from spreadsheets and other databases, shares information over Intranets and the Internet, and builds faster business solution. With the ODBC driver for Access, data can be retrieved from the database in client/server-based system.

# CHAPTER 4 SYSTEM DESIGN

## 4.1 System and Application Design

Software design is a process of devising and documenting the overall architecture for a software system. It includes identifying the major components of the system, specifying what they are to accomplish, and establishing the interfaces among the components. Design is the first step in the process of transforming the requirements into a close representation of the eventual function software. It also includes lower level work such as detailed specification of data structures and algorithms within the identified components.

In this project, the purposes of this phase are:

- To transform requirements into a working system.
- To determine a set of components and intercomponent interfaces those satisfy a specified set of requirements.
- To change the abstract logical model to the concrete physical implementation.

In system Development Life Cycle, the design phase is the stage where the requirement analysed in previous page phase, that is the System Analysis phase are translated into System Characteristics.

### 4.1.1 Design Consideration

In the stage, several aspects need to be considered such as:

- **Logical directory structure**

  A clear hierarchical directory structure makes the maintenance easier.

- **Effectiveness of user interface**

  An effective user interface is very important for a web application due to the fact that users are the key factors in determining the success of the system.

- **Size of database**

  It can affect the database performance. It should be implemented as accurate as possible.

- **Efficiency of process**

  A process should keep as simple as possible and the code should be efficient enough to maintain the responsiveness of the entire system at high level.

## 4.2 Process Design

This project design is based on data flow oriented design. It is also caned structured design. This type of design stresses on modularity, top-down design and structured programming.

- Structure Chart

  Structure chart is used to depict high level abstraction of a specified system. The use of structure chart is to describe the interaction between modules in a system.

- Module Description

  A module is a relative small unit of a system that is defined by its function. Modules are self-contained system components. As much as possible, all of the computer instructions contained in a module should contribute to the same function. Modules are executed as units and in most instances have a single point of entry and a single point of exit. For object-oriented programming languages, a module would roughly be a method. A computer program is typically made up of several modules. Modules may also represent separately compiled program, subprogram, or identifiable internal procedures.

## 4.3 Data Flow Diagram (DFD)

Data flow diagrams (DFD) depict the broadest possible overview of system inputs, processes and outputs. It able to conceptualise how the data moves through the organisation, the processes or transformation that the data undergoes, and what the outputs are [Kendall, 1999].

The data flow approach has four chief advantages over narrative explanations of the way data moves through the system. The advantages are:

1. Freedom from committing to the technical implementation of the system too early.
2. Further understanding of the inter-relatedness of system and subsystems.
3. Communicating current system knowledge to users through data flow diagrams.

Analysis of a proposed system to determine if the necessary data and processes have been defined [Kendall, 1999].

Figure 4.1 shows the DFD for servicing. After the customer key in their problems into our database, the employee will check the database for an update information. The employee will have to key in their password and the database will verified the user and give the user permission to view and update the database. After viewing at which customer having problems, the engineer will go to the company and service for them before update the table and close the case.

Figure 4.2 shows the DFD for product ordering. Same as the servicing, the employee checks the Order Table for update information. After checking the product need by the customer, the engineer will send the product to customer. If the branch didn't have the product needed, the administrator will transfer the product from other branch before sending them to the customer. Lastly, the user will update the table and close the case.

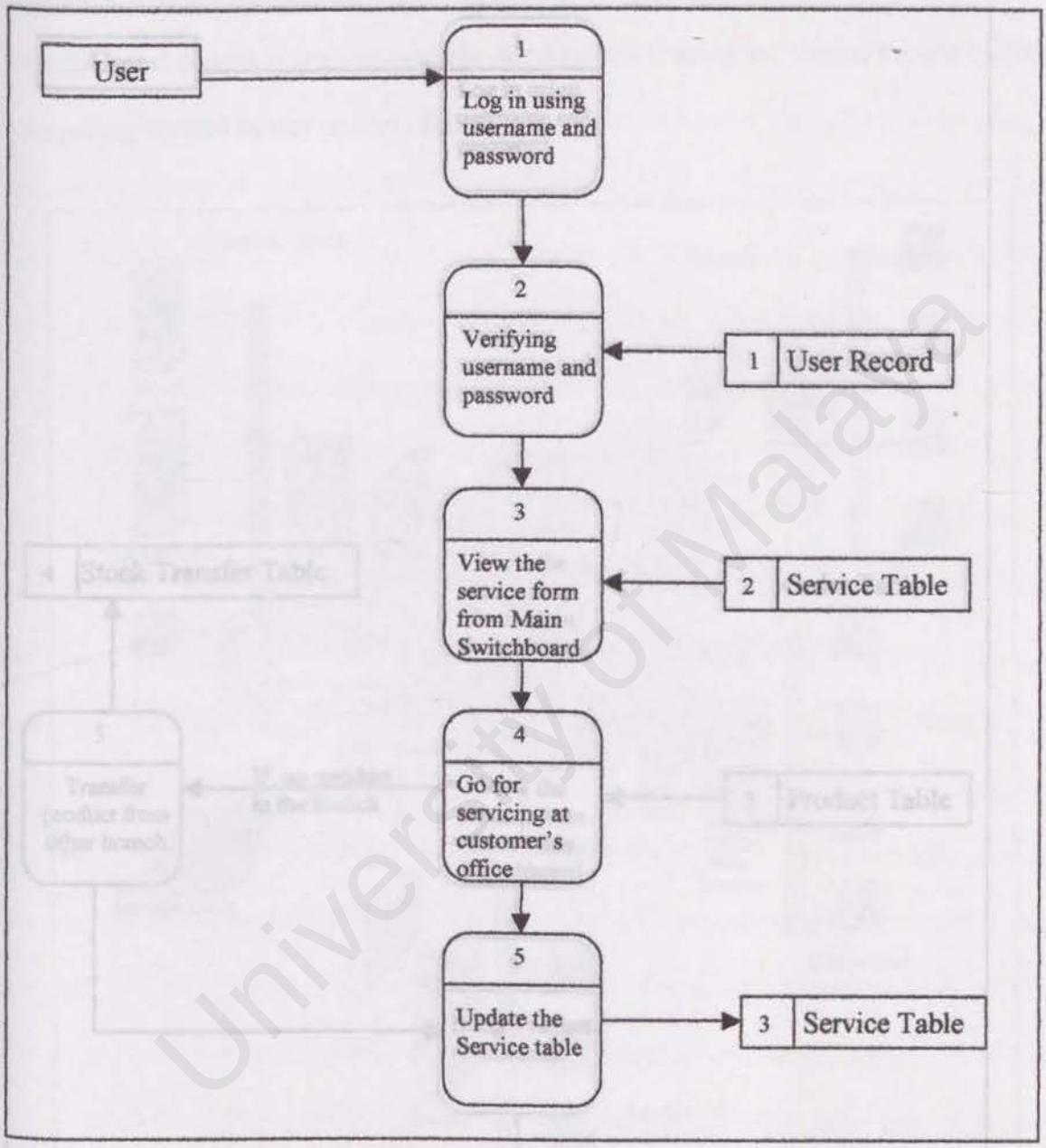


**Figure 4.1: DFD diagram for Servicing**

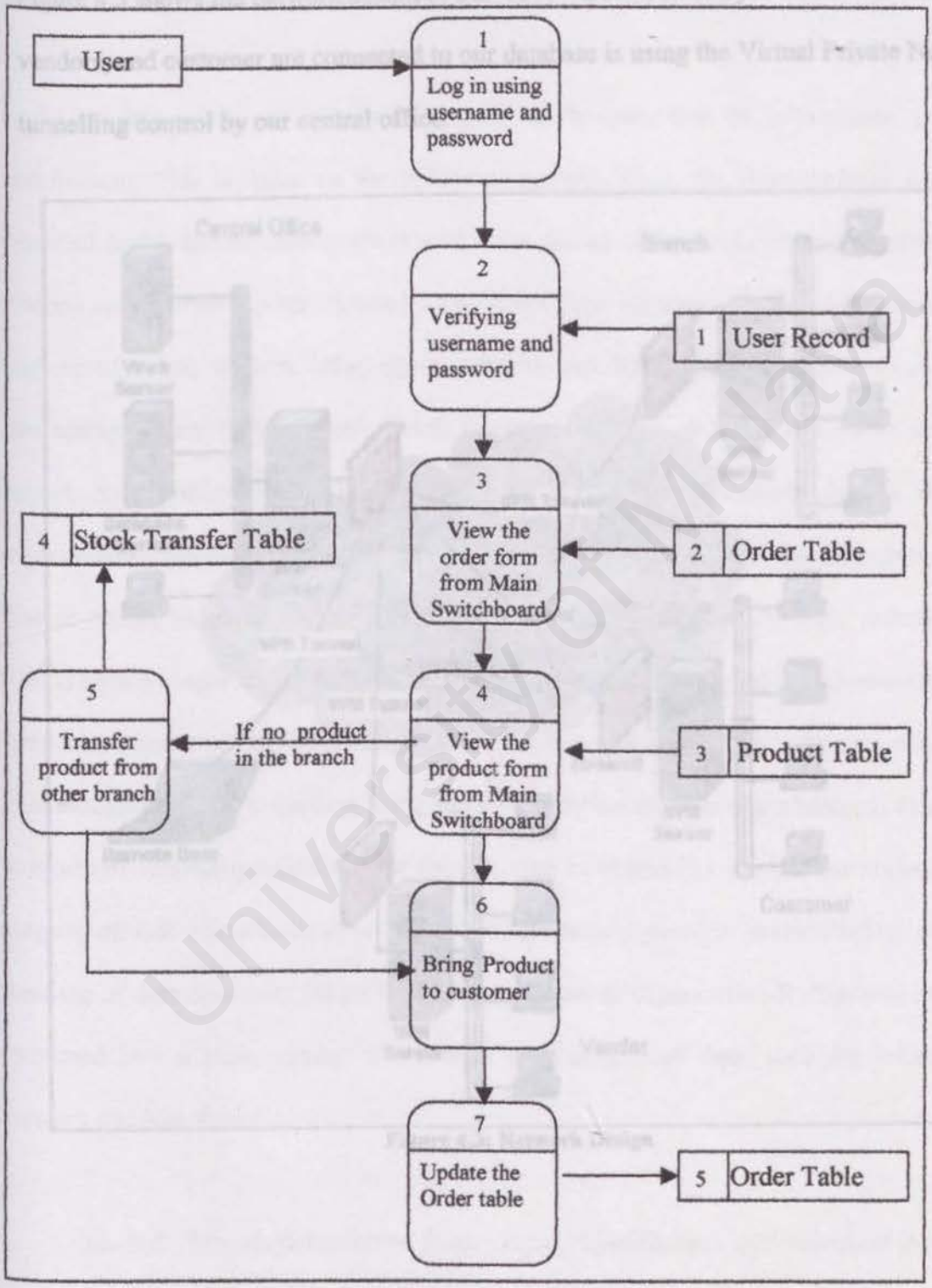


**Figure 4.2: DFD diagram for Product Ordering**

## 4.4 Network Design

Figure 4.3 shows the network design of our VPN – Data Center System. All the branches, vendors, and customer are connected to our database is using the Virtual Private Network tunnelling control by our central office.

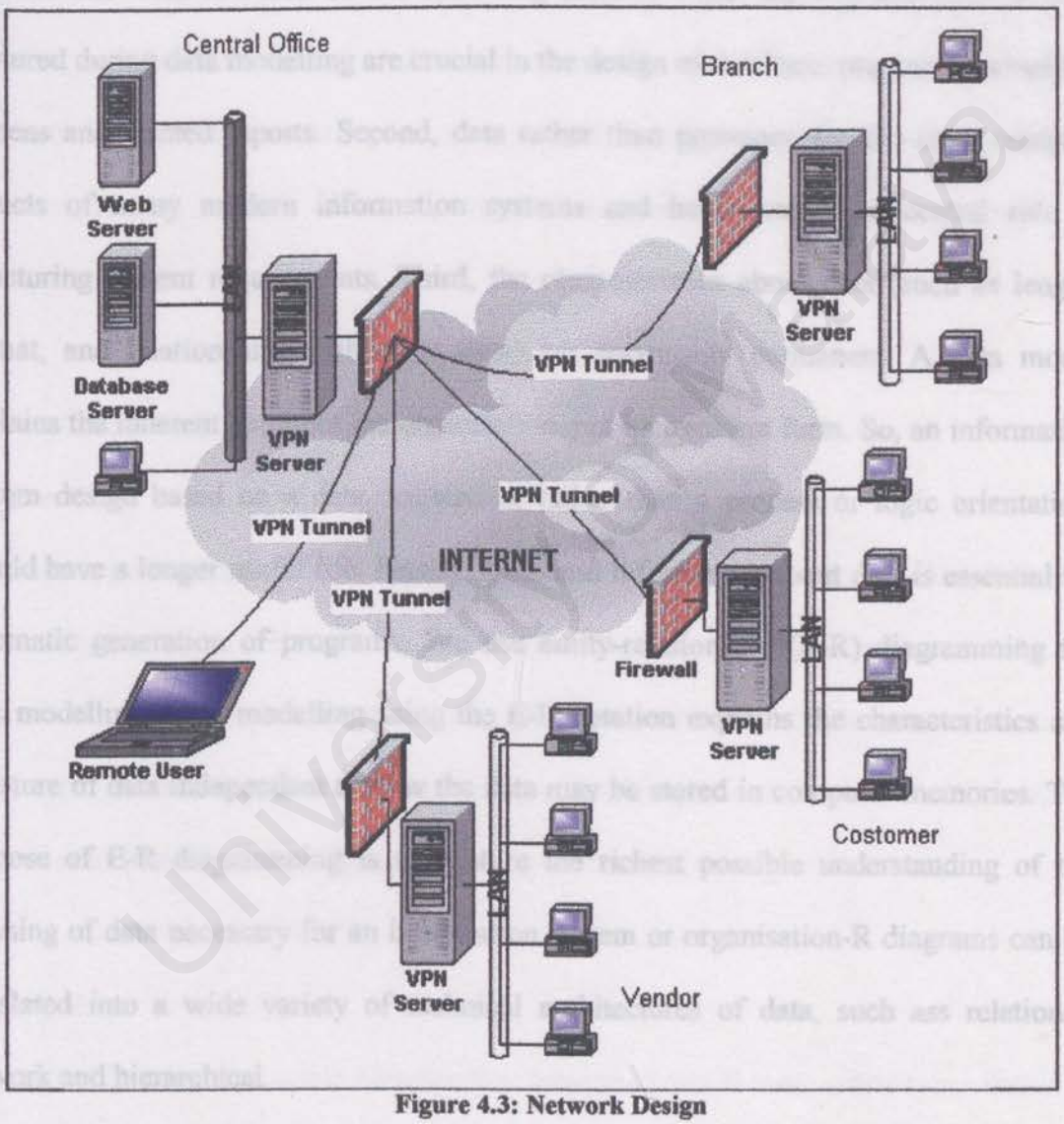


**Figure 4.3: Network Design**

## 4.5 Database Design

### 4.5.1 Database Design Using Entity Relationship (E-R) Models

A data model is an important part of the statement of information system requirement. This is based on the following reasons. First, the characteristics of data captured during data modelling are crucial in the design of database, programs, computer, screens and printed reports. Second, data rather than processes are the most complex aspects of many modern information systems and hence require a central role in structuring system requirements. Third, the characteristics about data (such as length, format, and relationships with other data) are reasonably permanent. A data model explains the inherent nature of the organisation, not its transient form. So, an information system design based on a data orientation, rather than a process or logic orientation, should have a longer useful life. Finally, structural information about data is essential for automatic generation of programs. We use entity-relationship (E-R) diagramming for data modelling. Data modelling using the E-R notation explains the characteristics and structure of data independent of how the data may be stored in computer memories. The purpose of E-R diagramming is to capture the richest possible understanding of the meaning of data necessary for an information system or organisation-R diagrams can be translated into a wide variety of technical architectures of data, such ass relational, network and hierarchical.

An E-R data model evolves from project identification and selection through analysis as it becomes more specific and is validated by more detailed analysis of system

needs. In the logical design phase, the final E-R model developed in analysis is matched with designs for systems inputs and outputs and is translated into a format from which physical data storage decisions can be made in the physical design phase. During physical design, specific data storage architectures are selected and then, in implementation, file and database are defined as the system is coded. Through the use of the project repository, a field in a physical data record can, for example, be traced back to the conceptual data attribute that represents it on an E-R diagram. Thus, the data modelling and design steps in each of the System Development Life Cycle phases are linked through the project repository.

The notation we will use for E-R diagrams is:

**Entities**

An entity type is a collection of entities that share common properties or characteristics. Each entity type in an E-R model is given a NAME. Since the name represents a class or set, it is singular. Also, since an entity is an object, we use a simple noun to name an entity type. We use capital letters in naming an entity type and ,in an E-R diagram, the name is placed inside a rectangle representing the entity.

**Relationship**

A relationship is an association between the instance of one or more entity types that is of interest to the organisation. Relationships are labelled with verb phase. The degree of a relationship is the number of entity types that participate in that relationship. The three

most common relationships in E-R models are unary (degree one), binary (degree two, and ternary (degree three).

- A unary relationship is a relationship between the instance of one entity type.
- A binary relationship is a relationship between instances of two entity types and is the most common type of relationship encountered in data modelling.
- A ternary relationship is a simultaneous relationship among instance of three entity types.
- The cardinality of a relationships is the number of instance of entity B that can (or must) be associated with each instance of entity A.
- The minimum cardinality of a relationship is the minimum number of instances of entity B that may be associated with each instance of entity A.
- The maximum cardinality of a relationship is the maximum number of instances of entity B that may be associated with each instance of entity A.

There are explicit links between a data model and a data flow diagram. Some important links are explained here. Data elements included in a data flows also appear in the data model and vice versa. You must include in the data modal any raw data captured and retained in a data store and a data model can include only data that has been captured or is computed from captured data. Each data store in a process model must relate to business objects (What we will call data entities) represented in the data model.

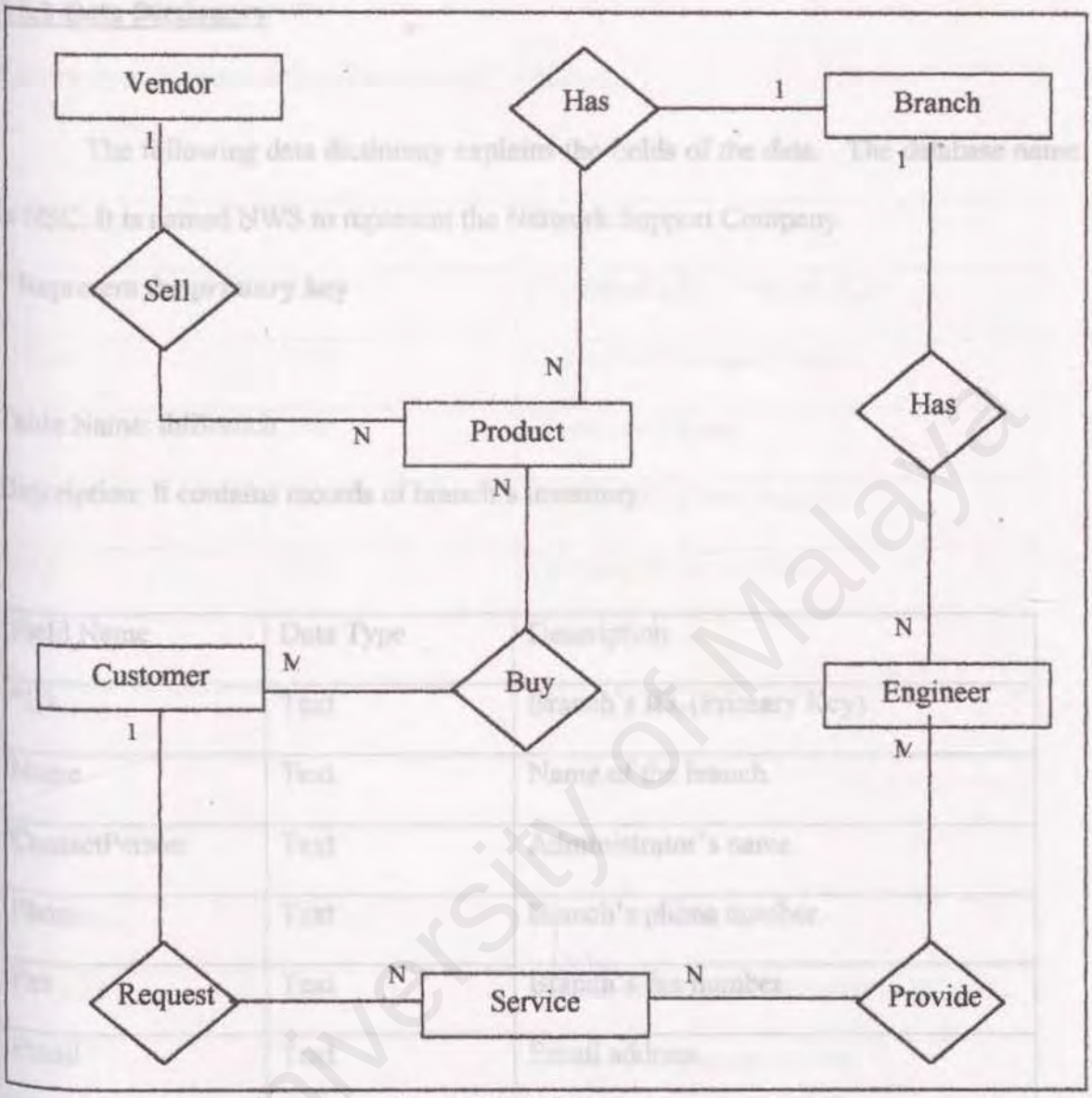


**Figure 4.4: E-R Diagrams**

| Entity | Relation |
|---|---|
| Vendor | One **vendor** can sell many **products** to the main company. |
| Customer | Many **customers** can buy many same or different **products**.<br>One **customer** can request many type of **servicing**. |
| Branch | One **branch** can store many **products**.<br>One **branch** can have many engineers. |
| Engineer | Many **engineers** can provide many types of **service** |

**Table 4.1: Entity Relation Description**

### 4.5.2 Data Dictionary

The following data dictionary explains the fields of the data. . The database name is NSC. It is named NWS to represent the Network Support Company.

* Represent the **primary key**

Table Name: tblBranch

Description: It contains records of branch's inventory.

| Field Name | Data Type | Description |
|---|---|---|
| *ID | Text | Branch's ID. (Primary Key) |
| Name | Text | Name of the branch. |
| ContactPerson | Text | Administrator's name. |
| Phone | Text | Branch's phone number. |
| Fax | Text | Branch's fax number. |
| Email | Text | Email address. |
| Address | Text | Branch's address. |
| City | Text | City name. |
| State | Text | State name. |
| PostCode | Number | Post Code of the area |
| Country | Text | Country where the branch is located. |
| Type | Text | To show they are branch |

**Table 4.2: Branch Table**

Table Name: tblCustomer

Description: It contains detail records of Customers.

| Field Name | Data Type | Description |
|---|---|---|
| *ID | Text | Customer's ID. (Primary Key) |
| Name | Text | Customer's company name. |
| ContactPerson | Text | Customer's name. |
| Phone | Text | Customer's phone number. |
| Fax | Text | Customer's fax number. |
| Email | Text | Email address. |
| Address | Text | Customer's address. |
| City | Text | City name. |
| State | Text | State name. |
| PostCode | Number | Post Code of the area |
| Country | Text | Country where the customer is located. |
| Type | Text | To show they are customer |

**Table 4.3: Customer Table**

Table Name: tblCategory

Description: It contains type of category.

| *Category | Text | Type of category |
|---|---|---|

**Table 4.4: Category Table**

Table Name: tblVendor

Description: It contains detail records of Vendors.

| Field Name | Data Type | Description |
|---|---|---|
| *ID | Text | Vendor's ID. (Primary Key) |
| Name | Text | Vendor's company name. |
| ContactPerson | Text | Vendor's name. |
| Phone | Text | Vendor's phone number. |
| Fax | Text | Vendor's fax number. |
| Email | Text | Email address. |
| Address | Text | Vendor's address. |
| City | Text | City name. |
| State | Text | State name. |
| PostCode | Number | Post Code of the area |
| Country | Text | Country where the vendor is located. |
| Type | Text | To show they are vendor |

**Table 4.5: Vendor Table**

Table Name: tblDesignation

Description: It differentiates the user type.

| Field Name | Data Type | Description |
|---|---|---|
| *Designation | Text | User type's (Primary Key) |
| Description | Text | Description |

**Table 4.6: Designation Table**

Table Name: tblUser

Description: It contains records of user account.

| Field Name | Data Type | Description |
|---|---|---|
| *UserID | AutoNumber | ID for record. (Primary Key) |
| Name | Text | Name of the user |
| Username | Text | Username for account. |
| Password | Text | User password. |
| CompanyID | Text | Company's ID. |
| Designation | Text | User type |
| ContactNumber | Text | User's Contact Number |
| Email | Text | User's Email Address |

**Table 4.7: User Table**

Table Name: tblTransfer

Description: It contains records of request to move back stock from branch.

| Field Name | Data Type | Description |
|---|---|---|
| *TransferID | AutoNumber | ID for records. (Primary Key) |
| Date | Date/Time | Date request issued. |
| ProductName | Text | Name of the product. |
| FromBranchID | Number | Branch's ID. |
| ToBranchID | Number | Branch's ID. |
| Amount | Number | Amount of product requested to move. |

**Table 4.8: Transfer Table**

Table Name: tblProduct

Description: It contains detail records of products.

| Field Name | Data Type | Description |
|---|---|---|
| *ProductID | AutoNumber | Id of the product (Primary Key). |
| ProductName | Text | Name of the product. |
| Category | Text | Category. |
| BuyingPrice | Currency | Dealer prices of the product. |
| SellingPrice | Currency | Selling price of the product. |
| Description | Text | The description of the product. |
| Date | Date/Time | Order date. |
| B001 | Number | Number of stock in branch 1 |
| B002 | Number | Number of stock in branch 2 |
| B003 | Number | Number of stock in branch 3 |
| QuantityIn | Number | Number of stock order. |
| QuantityOut | Number | Number of stock sold out. |
| Balance | Number | Number of stock left. |

**Table 4.9: Product Table**

Table Name: tblOrder

Description: It contains detail records of order post up to our company.

| Field Name | Data Type | Description |
|---|---|---|
| *OrderID | AutoNumber | Order ID. (Primary Key) |
| DateOrder | Date/Time | Date when the product is ordered. |
| ProductName | Text | Name of the product. |
| Category | Text | Product Category. |
| Amount | Number | Amount of product ordered. |
| BranchID | Text | Branch's ID that send product. |
| CustomerName | Text | Customer's company name that order product |
| CustomerID | Text | Customer's ID that order product. |
| DateSent | Date/Time | Date when the product is sent. |
| UnitPrice | Currency | Price per Unit. |
| TotalPrice | Currency | Total price of the transaction. |
| Status | Text | To know whether the order have been send. |

**Table 4.10: Order Table**

Table Name: tblService

Description: It contains records of service order posted to our company.

| Field Name | Data Type | Description |
|---|---|---|
| *ServiceID | AutoNumber | Service's ID. (Primary Key) |
| Problems | Text | Problems description. |
| Solutions | Text | Solutions description. |
| BranchID | Text | Branch's ID. |
| CustomerName | Text | Customer's company Name |
| CustomerID | Text | Customer's ID. |
| AcceptBy | Text | Employee's ID. |
| LoggedDate | Date/Time | Date when the service is requested. |
| BeginDate | Date/Time | Date when the service is provided. |
| ClosedDate | Date/Time | Date when the service is completed. |
| Status | Text | Finished or not. |

**Table 4.11: Service Table**

## 4.6 User Interface Design

User interfaces is not easy task. The first step is to define the overall look and feel of the site. Then the flow of what a user will fee and see have to be determined. Navigate links need to be provide on all pages so that the user can more between different pages easily. There are several general principles for user interface design, which are consistency, recoverability, confirmation and verification message, responsiveness and reverse action.

The step in designing on effective user interface are:

1) Define a purpose for the interface
2) Identify the users expectation and needs
3) Design user interface
4) Conduct usability testing

# CHAPTER 5 SYSTEM IMPLEMENTATION AND CODING

## 5.1 Introduction

After the system design, the next step is the implementation of system. The system implementation is divided into two parts, that is that system hardware and software. Nearly all the design phases that have been presented to this point are directed towards a final objective: to translate representation of software into a form that can be "understood" by the computer. The primary goal of this phase is the production of a simple, clear source code with internal documentation that will ease the processes of verification, debugging, testing, modification and further enhancement

## 5.2 System Hardware

For VPN, we need a server to store the database so that the system can access the database at any workstation by using the Internet line. To access the server, every workstation is needed to have modem to access truth the Internet to the server. As this is a system using Microsoft Access, the program needed a large sum of memory spaces to store the database. The speed of the processor is needed to reduce the pop-up time and the closing time of the forms while running the system. Other hardware requires is mouse and keyboard, as the basic hardware using in the windows environment.

## 5.3 System Software

| Software | Purpose | Descriptions |
|---|---|---|
| Microsoft Windows 95 | System requirement | Operating system |
| Microsoft Visual Basic 6.0 | System requirement | System design and coding |
| Microsoft Access 97 | System requirement | Database design |

**Table 5.1: Summary of Software Tools Used**

Microsoft Visual Basic 6.0 has been chosen to develop the system because of the following reason:

### 5.3.1 Windows Programming

Microsoft Visual Basic 6.0 practices the windows programming approaches. Users are able to experience the system such as the point and click function and multitasking in the window environment. Whereas programmers' works progress become efficient because of the independent graphics and predefine functions or objects.

### 5.3.2 Object-Oriented Programming (OOP)

Object-Oriented Programming encapsulates data and functions into packages. The data and functions of an object are intimately tied together. Object will hide the information of the implementation details within themselves. The advantage of applying the OOP is it enables function reusability and easier maintenance.

### 5.3.3 Event-Driven Programming

Microsoft Visual Basic 6.0 has an event-driven programming environment. It is the quickest way to create powerful application under Microsoft Windows operating system. Events determine the control's reactions to external event. Button, textbox, combo box, label, and so forth in Microsoft Visual Basic 6.0 are called controls. Events recognised by the various controls but are handled by the application. A command button will respond according to the event coded once the user clicks on it. In window applications, mouse events occupy a big proportional of the possible events triggered by the user such as click, move and drag. Microsoft Visual Basic 6.0 controls handle mouse events very well.

### 5.3.4 Error Handling

Microsoft Visual Basic 6.0 provides an integrated debugger. Debugging is the process to remove some part of the program that does not work properly. Error handling can be implemented efficiently in Microsoft Visual Basic 6.0. Error handling is used to handle unexpected events in the program. Prompts and messages are displayed to deliver useful information about errors encountered in the program to the programmer.

## 5.4 System Security Control

The security control is needed for every system. This is to prevent the system from criminals and vandals. For security control, the system has provided the authentication log on for secondary security to ensure only the user with the correct username and password can use the system. Beside that, the data transfer from and to the database in the server has been protecting by the VPN using encryption base in the server.

## 5.5 Program Development

Program development is the process of creating the programs needed to satisfy an information system's processing requirements. Program development consists of the following 5 steps: review the program documentation, design the program, code the program, test the program and completion the program documentation.

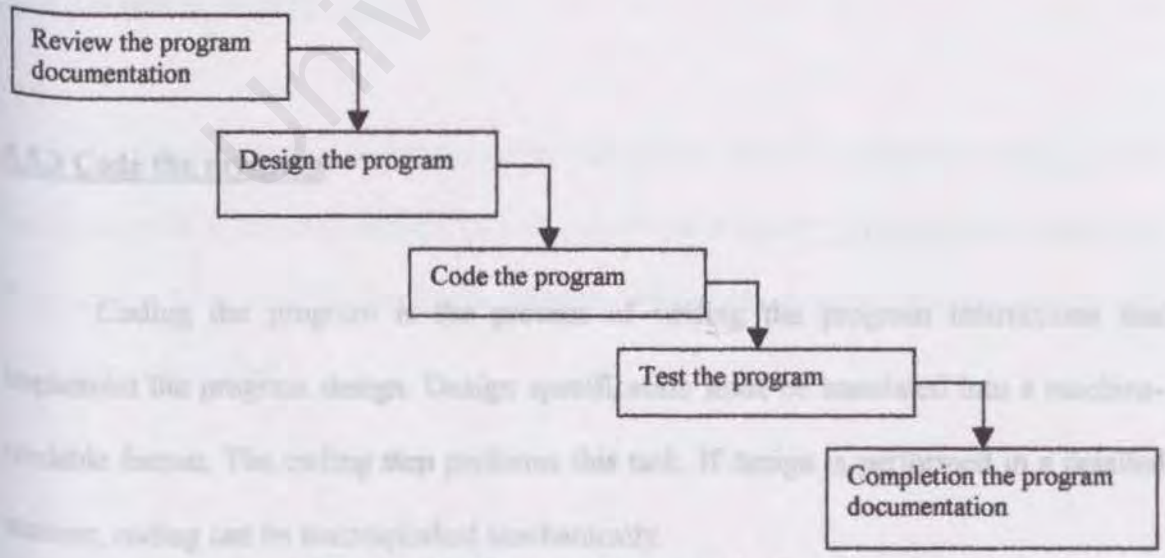


**Figure 5.1: The five steps of program development**

### 5.5.1 Review the program documentation

The first step in the program development is to review the program documentation that was prepared during the previous phases. The program documentation of catalogue ordering system consists of simple process descriptions, report layouts, data dictionary entries and the source documents. This documentation helps me to understand better the work that needs to be covered during this coding phase.

### 5.5.2 Design the program

After the program documentation review, I need to design the program, which is the second level of program design during the system development. For this second level of program design, I have exactly decided how the program can accomplish what it must do by developing a logical solution to the programming problems. The logical solution, or logic, for a program is a step-by-step solution to a programming problems.

### 5.5.3 Code the program

Coding the program is the process of writing the program instructions that implement the program design. Design specification must be translated into a machine-readable format. The coding step performs this task. If design is performed in a detailed manner, coding can be accomplished mechanically.

### 5.5.4 Test the program

During the testing program level, I must thoroughly test a program to ensure it functions correctly before the program processes actual data and produces information on which people will rely. I will perform several types of test on an individual program. (Will be further discusses in details in section below).

### 5.5.5 Document the program

Accurate and complete program documentation is essential for the successful operations and maintenance of the information system. This documentation includes the system user manual that may needed by most of the customers as well as the system administrator's.

## 5.6 Program Coding

### 5.6.1 Coding Approach

A program with a technique called top-down, stepwise refinement, an approach that is essential to the development of well-structured program. This approach enables the programmer terminates the top-down, stepwise refinement process when the pseudocode algorithm is specified.

### 5.6.1.1 Coding Style

Coding style is an important attribute of source code and it determines the intelligibility of a program. An easy to read source code makes the system easier to be maintained and enhanced. The elements of coding style include internal (source code level) documentation, methods for data declaration and approach to statement construction.

### 5.6.1.2 Code Documentation

Code documentation begins with the selection of identifier (variable and labels) names, continues with the composition of connectivity and end with the organization of the program. Use blank line or indentation so that comments can be readily distinguished from code.

**i) Internal documentation**

Internal comments provide a clear guide during the maintenance phase of the system. Statements of purpose indicating the function of the module and a descriptive comment that is embedded within the body of the source code is needed to describe processing functions.

**ii) Naming convention**

Naming convention provides easy identification for the programmer. The naming convention as created with coding consistency and standardization in mind.

**iii) Modularity**

In order to reduce complexity, facilitate change results in easier implementation by encouraging parallel development of different parts of a system.

# CHAPTER 6 TESTING

Under this stage, we perform many type of testing to ensure that the final system perform as what it should be. Many type of testing have to be done before the system is released to the customer to ensure that the system is developed according to its specification and every function implemented in a program works correctly. Testing is a critical element of software quality assurance and represents the ultimate review of specification, design and coding. Rules that can serve well as testing objectives are:

- Testing is a process of executing a program with the intent of finding an error.
- A good test case is one that has a high probability of finding an undiscovered error.
- A successful test is one that uncovers an as yet undiscovered error.

Four types of testing are being used for the database; there are unit testing, module testing, integration testing and system testing. The following are the testing that was carried out under this stage.

## 6.1 Unit Testing

We use the unit testing technique to ensure that the stand-alone program fixes the bug without side effects. After one new module is developed, we usually test it independently in order to assure their accuracy and to find faults in the modules. For this system, unit testing was done during the coding phase. The first step is to examine the

program code by reading through it, trying to spot algorithm, data and syntax faults. This is followed by comparing the code with specifications and with the design to make sure that all relevant cases have been considered. Finally, test cases are developed to show that the input is properly converted to the desired output.

There are three kinds of testing strategy carried out for the unit testing. The following sessions explain the testing strategies that were carried out throughout the project.

### 6.1.1 Code Reviewing

Under this strategy, we review the codes. The codes are examined line by line in order to make sure that many uncovered semantic errors during implementation could be revealed. In reviewing the code, the correctness of coding was identified by comparing it to the original design of the program flow. When the logic and flow of the program were identified, the code was commented so that it can be traced in the future.

The code was also examined and debugged in order to identify any fault coding. It is easier to debug the error using Visual Basic if compared to the ASP coding. We can trace the Visual Basic code line by line using the debugger available. After the testing, the final system is in accordance with the system specifications.

### 6.1.2 Test Cases

Besides reviewing the codes, we use some test cases to test the system. This approach is used as some set of structural input is given and output is observed. This strategy is needed to identify the variance between the prototype and the requirement. In this testing, we input different of data to the program. For example to test the authenticate logon module, we input different login ID and password to test the program. With this, the reaction of the program to the input data could be tested. This could identify the program's faults, which probably happen in normal condition.

### 6.1.3 Other Users

After the two testing has been used, we launch the beta version of the system to other users for testing purpose. This is to identify the fault that may incur in any other unexpected condition. The testing involved with random data in random situation. From the testing, we get some feedback from the user. This feedback provides some important information about the usability and reliability of the application.

## 6.2 Module Testing

After the unit testing, we perform the module testing which include the user module testing and the administrator module testing. We create one administrator and some user data in performing the testing. The testing was carried out to ensure that the

codes under the module function accordingly when all units of code are integrated. Some module like the updating attendance module is test together with the attendance clock in or clock out module because the module has similar functionality. If the error is present from a particular module, the part of the module that goes wrong is identify and unit testing is used to identify the errors.

## 6.3 Integration Testing

After all the modules were believed to have satisfied the requirements, they were integrated with the Leave Management System. During the integration, the testing was carried out in order to identify the fault and failures caused by the integration as well as review and rectify the correct path of the system flow.

During the integration, all the module prototypes were combined and tested in a testing environment. The testing environment was consistent for all the modules in terms of interface, user authentication and function calling procedures. The program flow and the testing needs for each of the modules were reviewed and identified. Then, the program flow for the entire system were reviewed and tested. After that, the entire system was tested with some test cases.

Incremental integration approach was applied during the developments of the system. The system was constructed and tested in small arguments, where errors were easier to isolate and correct.

## 6.4 System Testing

Finally, we perform the system testing to ensure that the entire application, of which the modified program was a part, still works. It is used to test the integrated system and verify whether it meets the specified requirements. Others member of the project have helped us perform this kind of testing.

System testing is designed to reveal bugs that cannot be attributed to individual component, or to the interaction among components and other objects. System tests study all the concerns issue and behaviours that can only be exposed by testing the entire integrated system or major part of it. Catalogue Ordering System undergone three types of testing:

i) **Security Testing**

Verify the protection mechanism in the system against improper penetration.

ii) **Stress Testing**

Stress test is to determine whether a program fulfil the requirements defined for it. Equally important is to make sure that program works as it should, even under extreme condition.

**iii) Performance testing**

It is designed to test the run-time performance of system within the context of an integrated system. It occurs through all steps in the testing process.

## 6.5 Analysis of Test Results

From the testing process that has been carry out, it can summarized the test results as follow:

*i. Achieve the main objectives of the project*

Generally, the main objectives of the project as described earlier have been achieved. The system can handle and maintain the customers, suppliers, employees and product databases. It is a system, which is able to handle the purchasing and selling procedures. This is an important and major activity in a business organization. For the management, the various types of reports have been generated. Besides it provides safeguard to prevent the unauthorized users to access or modify the system / database.

*ii. Enhancement on the user interfaces*

The user interface for the system should be more attractive and user-friendly in order to attract the user to use the system. As some of the user may not computer-literate,

so it's important to provide the user interface as easy to use as possible. The customers may reluctant to use a system that is not user-friendly. So, using of graphics or more attractive icons to represent the buttons may help to improve the user interface.

*iii. Enhancement on the product information module*

Since this is a catalogue ordering system, so the product information must be clear. This includes the picture and description of the product and also its functionality. This information not only can give the overall clear picture for the customer but also can convince them for the product.

# CHAPTER 7 SYSTEM EVALUATION

## 7.1 Introduction

System evaluation is the post-implementation review to determine strengths and limitations/constraints of the system. The appraisal will provide feasible information to enhance the future project. It also highlights on the knowledge obtained and identifies shortcoming encountered in the system development and steps to be taken in solving problems.

## 7.2 System Strength

### 7.2.1 Window Platform

The system is designed to operate on the window platform, which is the most popular and widely used operating system among the end users. Thus, it can be easily adapted to most personal computers and users. The system is logically organized in a system manner that enables the users to operate smoothly.

### 7.2.2 User Friendly Interfaces

The system is considered as user friendly and data integrity. Its interfaces are intuitive where the users can easily get accustomed to the system's concept-based style. Graphic User Interface (GUI) components such as command buttons, combo box and navigation button are used to minimize the user actions while performing certain task.

The learning curve is foreseen to be short and a user should be able to use the system within minutes.Normally, users will not face problems when using the system although they are computer illiterate.

### 7.2.3 Security Control

Only authorized users are allowed to access to the system. They can order, submit problem, or update the data stored in the database.

### 7.2.4 Report printing

The system allows the administrators to view the reports and the reports can be printed for analysis purpose. The reports contain a critical as well as important data that required by the higher level management. It also can function as to evaluate the performance of the organization in certain area such as delivery and sales performance within a certain periods of times. So, printed reports are always helpful especially for analytical purpose.

## 7.3 System Limitation

Despite of the strengths, there are some limitations in the system:

### 7.3.1 Speed of form loading

The processing power of the system depends on the main storage of the personal computer used. If the system is run on a personal computer with 16MB RAM, the form loading process is quite slow. In order to run the system smoothly; it is advisable to run it on a PC with at least 32MB RAM.

### 7.3.2 Less Useful Reports

Even though the report is provided, but it can still be improved by providing more useful and meaningful reports for the management. Reports such as top ten best suppliers and stock replenishment report are useful to the high level management in the organization to do evaluation, projection and analysis for future planning.

## 7.4 Future Enhancement

The system needs further enhancement to improve its performance. Steps that are considered to be part of the system future enhancement are:

### 7.4.1 Payment Module

The system should be able to cover the payment module. This module should provide accurate and timely payroll or payment processing. This module is very important to be added into the system before the system can consider (basically) as a

complete system. After this module has been added, the administrator can make sure that the total amount owned by a particular customer cannot exceed certain volume/amount.

### 7.4.2 Backup and Restore function

The backup and restore function was not considered earlier due to initial functional requirements. This function is very important if an accident occurred causing damage to the system and database. So, the contingency planning is needed because the disaster will cause loses to the organization. The backup must be schedule in one proper manner that the important and critical data must backup in daily basis.

## 7.5 Problems and Solutions

During the implementation of this system, several problems had been encountered. The problems and solutions are as follows:

### *i. Processing Speed of PC*

At the initial stage, the application of the system was executed in a PC with 16MB RAM and the system can still run smoothly. As more and more forms and graphics added in the system, its performance was affected. The system unable run at the normal speed and become very time-consuming.

**Solution**

The PC was upgraded to 64MB RAM to solve this problem.

### *ii. Ambiguity in choosing development tool*

Choosing a right tool to develop the system was a difficult task because of inexperience and unfamiliarity with the new tool functions.

**Solution**

Discussion with fellow course-mates and self-study had given me some guidelines and assistance in choosing the development tool for the system.

### *iii. Difficult to find discussion members*

Many course-mates used different development tools to implement their system. It was difficult to ask for their advice and opinions when facing problems in the implementation phase. They were not familiar with the tools.

**Solution**

Most of the time this problem is overcome by referring to reference books.

## 7.6 Knowledge and Experiences Gained

Throughout the development of the system, knowledge and experiences have been acquired. The benefits are listed as below:

i. A golden opportunity to learn additional programming language, such as Visual Basic.
ii. Have the hands-on experience to plan and develop a system.
iii. Improve the skills in time management.

iv. Have a chance to know and use several software, which are not familiar before.

v. Ability to work independently and work under pressure.

vi. Enrich the experience in problem solving.

vii. Learn to handle a project.

viii. Learn the skill in writing documentation.

## 7.7 Conclusion

VPN-Data Center System has finally completed and had fulfilled the objective and scope established. It provides security to protect the data transfer trough Internet with encryption and description. The visual effects of the system make the learning process more interesting and stimulating. Owing to the limitations, future enhancement is recommended for improvement and development of the system. One of the most essential knowledge and experiences gained during this project is to learn the way of handling a project

In my opinion, as undergraduates should appreciate this learning process which gives us the opportunity to develop and improve our skills in designing, implementing and operating a project successfully under the guidance and supervision of the lecturer. By reviewing to the overall results, it can be concluded that the outcome of VPN-Data Center System has achieved and fulfilled the project objectives as set completely. The project is indeed a technical and economical feasible system for education purpose.

## Bibliography


1. Pfleeger, Shari Lawrence, "Software Engineering Theory and Practice", Prentice Hall, 1998.
2. Cowart, Robert, "Windows NT Server 4", Sybex, 1997.
3. Wynkoop, Stephen, "Special Edition Using Microsoft SQL Server 7.0", QUE, 1999.
4. "Microsoft Web Site: Window NT" http://www.microsoft.com/iis/default.asp.
5. Kendall, K.E. and Kendall, J.E. 1999. "Systems Analysis and Design". 4th Edition. Upper Saddle River, N.J., Prentice-Hall Inc.
6. David M. Kroenke, 6th Edition, "Database Processing, Fundamentals, design and implementation", Prentice-Hall Inc.
7. Cary N. Prague, Certified Access Developer, & Micheal R. Irwin, "Access 97 Bible", Ed. 1st, Foster City, CA IDG books Worldwide, Inc. 1997.
8. Brian Siler and jeff Spotts, "Special Edition Using Visual Basic 6", QUE, 1998.
9. http://www.compatibale.com/vpn now/education.html.
10. Charlie Scott, Paul Wolfe and Mike Erwin. "Virtual Private Networks". O'Reilly & Associates, 1999.
11. Bruce Perlmutter. "Virtual Private Networking". Prentice Hall PTR, 2000.
12. Teresa Canady, Pete Harris, Susie Parrent. [1999]. "Microsoft Visual Basic 6.0 Development". Microsoft Press. pp 4.
13. Igor Hawryszkiewycz, 1998. "Intruduction to Systems Analysis and Design, 4th Edition. Prentice Hall Australia Pty. Ltd.

# APPENDIX A: USER MANUAL

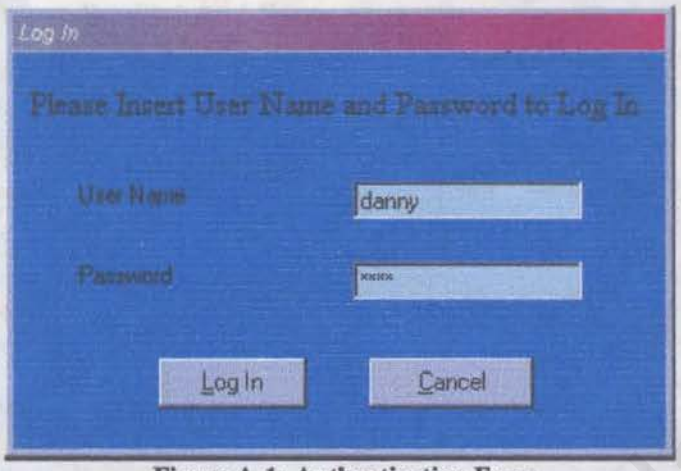


**Figure A-1: Authentication Form**

- ***User Name*** – To key in user name
- ***Password*** – To key in password for authentication
- ***Log In button*** – To log into the program after key in username and password
- ***Cancel button*** – To close the authentication form and get out of the program

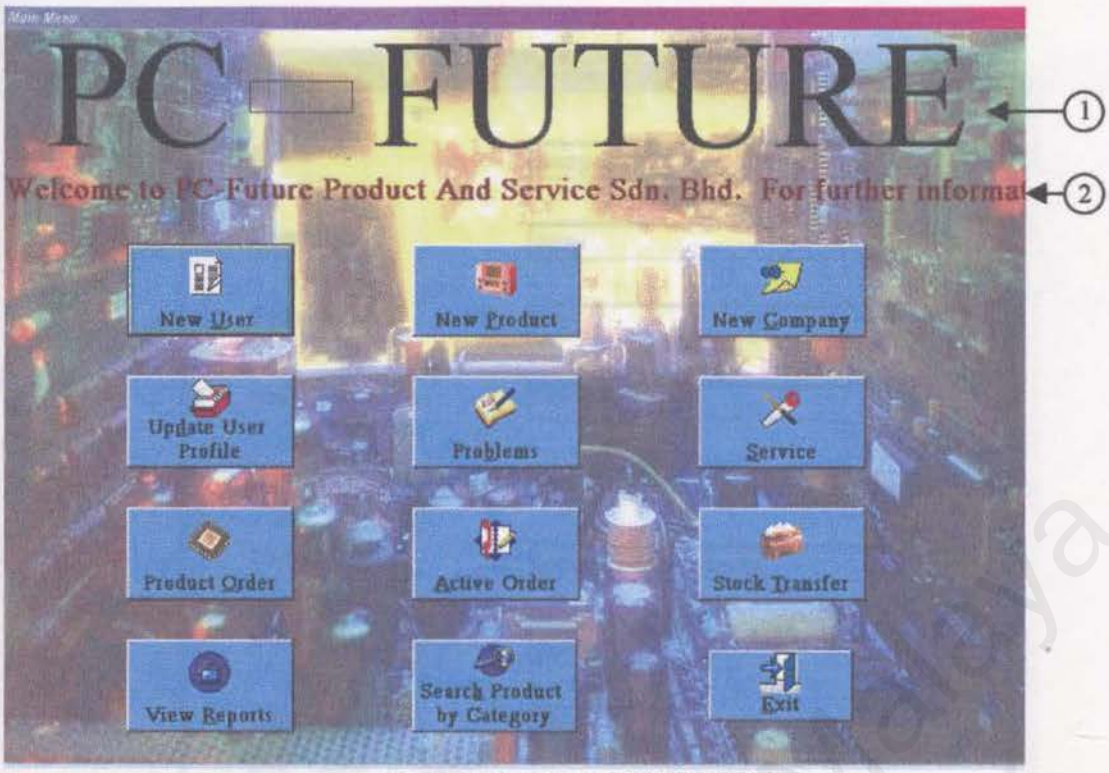


**Figure A-2: Program's Main Menu**

① – The Company's name

② – The company's banner which will be moving from Left to Right

***New User button*** – To sign up new user to use this program

***New Product button*** – To key in the products that the company buy from vendors

***New Company button*** – To record the particulars of the customers' and vendors' company

***Update User Profile button*** – To update user's profile

***Problems button*** – For customer to key in their computer problems.

***Service button*** – For administrator to view the problems faces by customer which hasn't been solve

***Product Order button*** – For customer to order products

***Active Order button*** – To view orders that haven't been send.

***Stock Transfer button*** – to key in the stock that has been transfer from one branch to another branch

***View Reports button*** – to view and print reports

***Search Product by Category button*** – To search product that the company sells

***Exit button*** – To exit the program

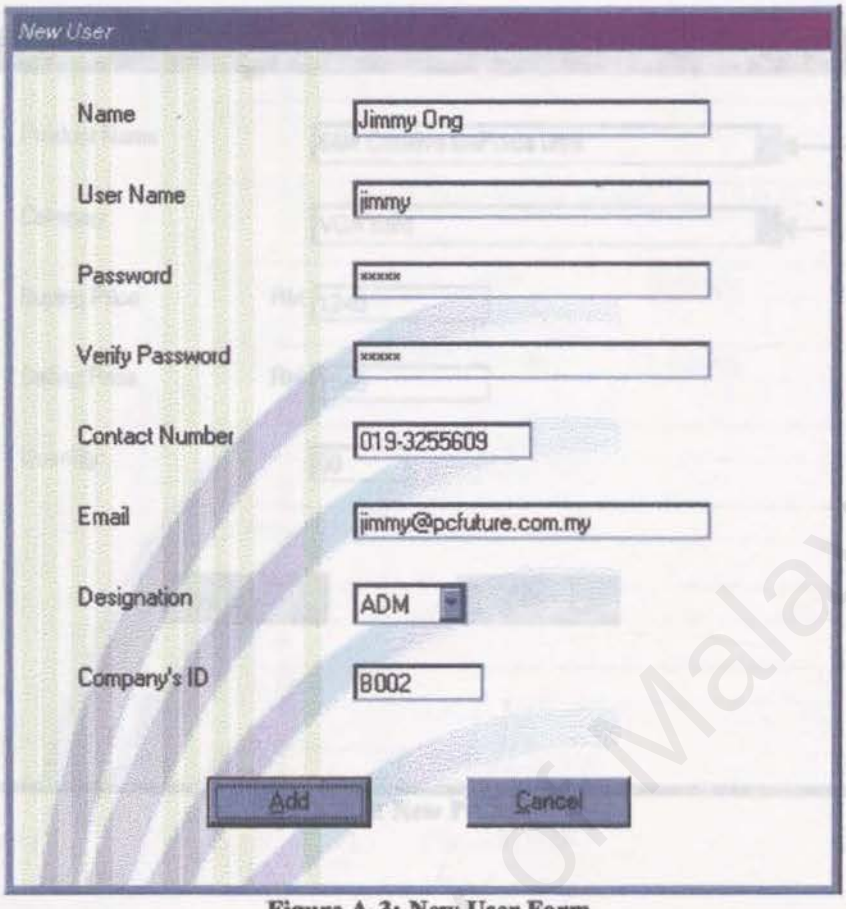


**Figure A-3: New User Form**

***Add button*** – Add new user to use the program

***Cancel button*** – To close New User Form without adding new user and return to Main Menu

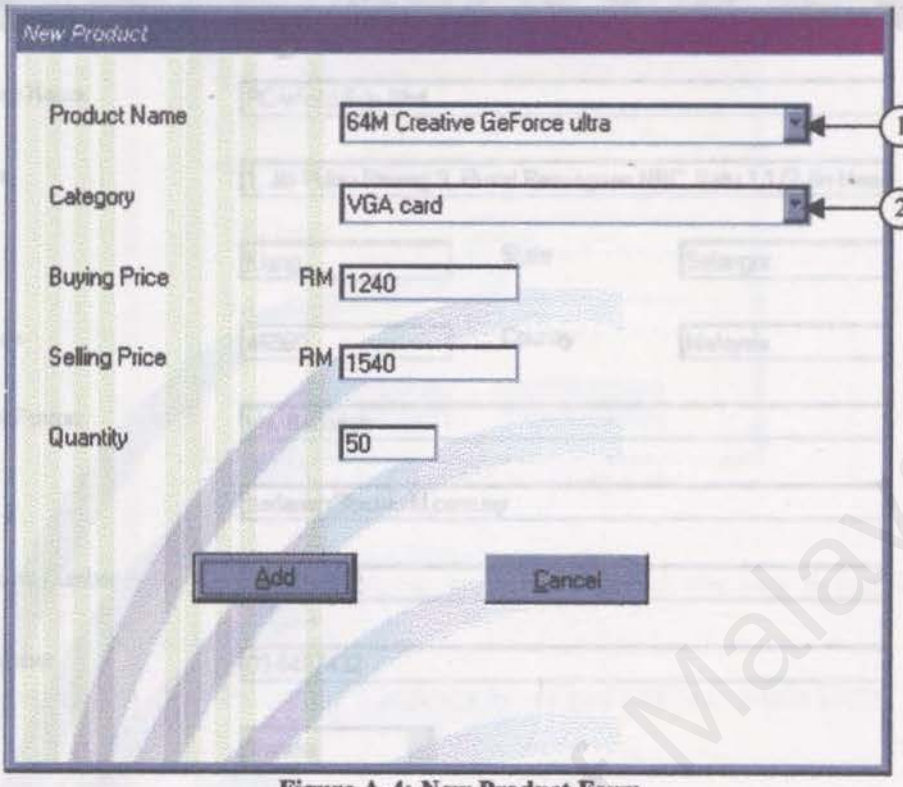


**Figure A-4: New Product Form**

① – Click to search the name of product that the company buy before

② – Click to insert product's category if the product is new

***Add button*** – To add the details and the quantity of the product that the company buy from vendors

***Cancel button*** – To close New Product From without adding any information to the database and return to Main Menu

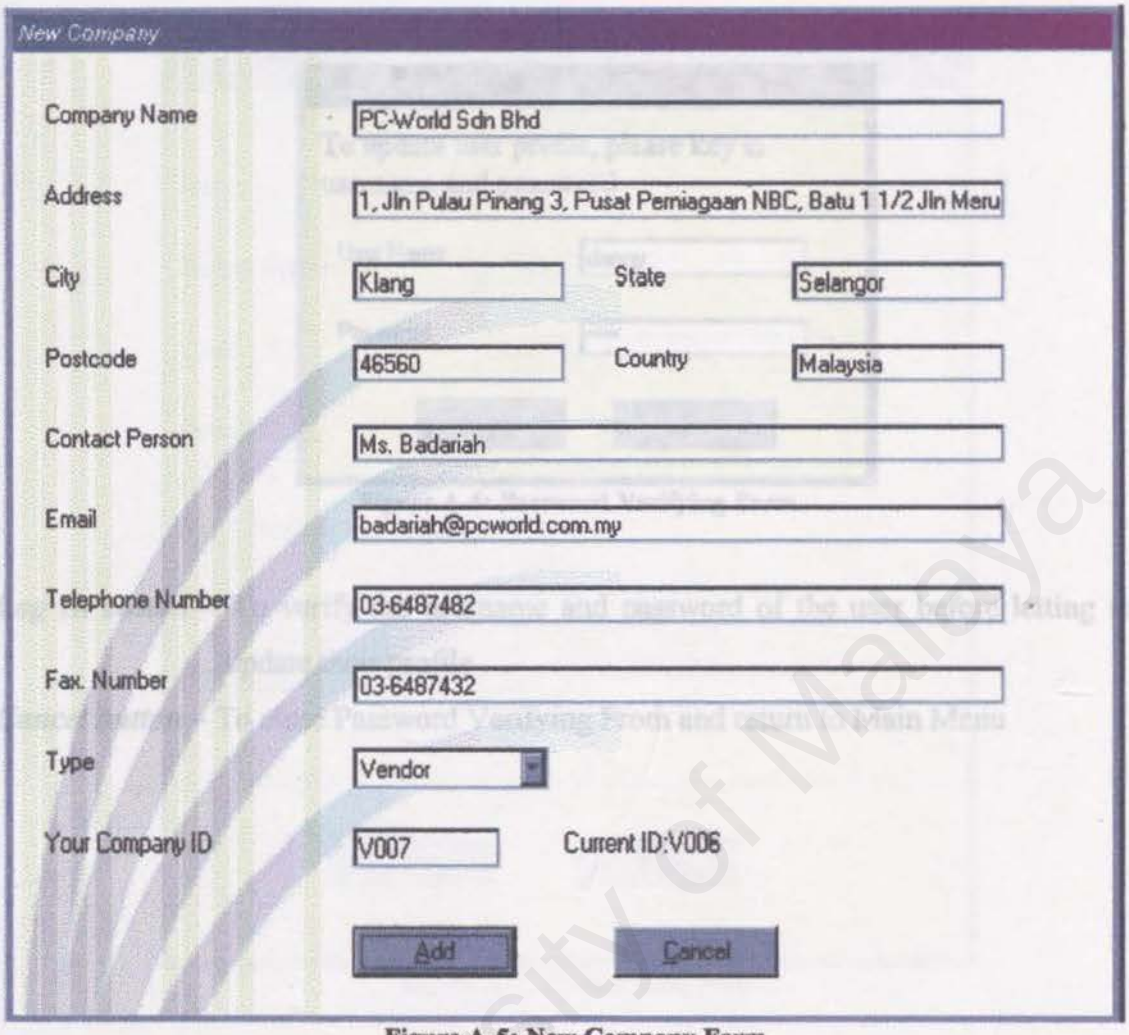


**Figure A-5: New Company Form**

***Add button*** – To add new vendors' and customers' company that our company corresponds with

***Cancel button*** – To close New Company From without adding any information to the database and return to Main Menu

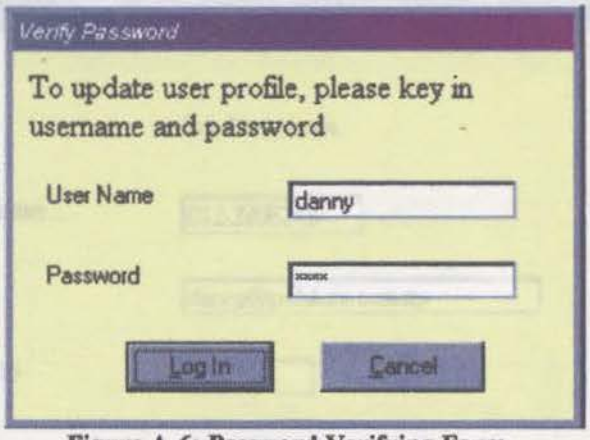


**Figure A-6: Password Verifying Form**

***Log In button*** – To verify the username and password of the user before letting user update their profile

***Cancel button*** – To close Password Verifying From and return to Main Menu

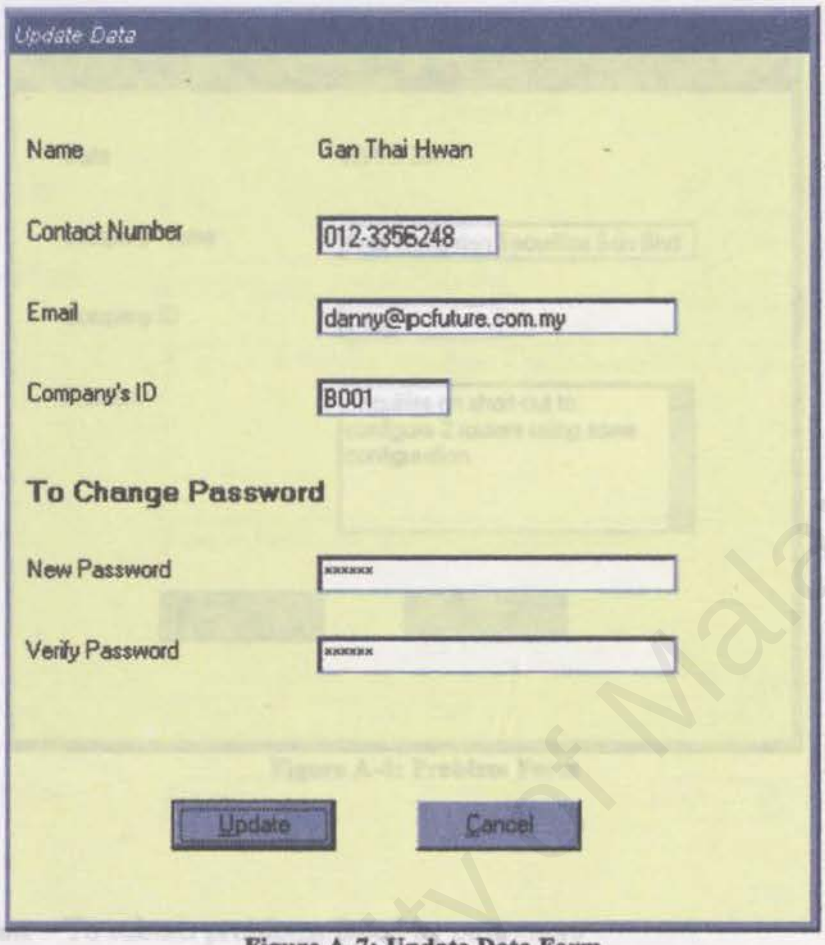


**Figure A-7: Update Data Form**

***Update button*** – To update user profile

***Cancel button*** – To close Update Data From without updating user information and return to Main Menu

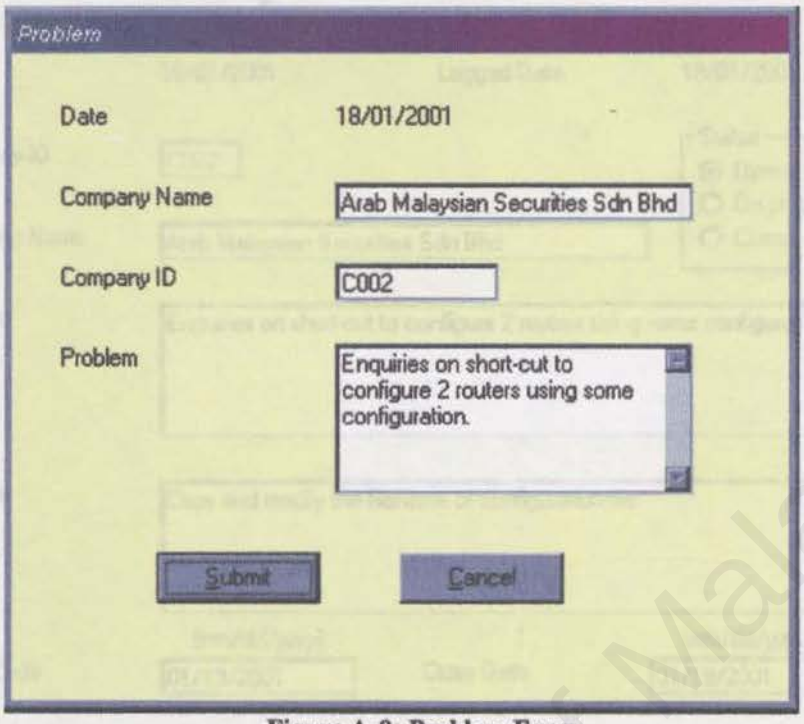


**Figure A-8: Problem Form**

***Submit button*** – To submit problems faced by customers

***Cancel button*** – To close Problem From without submitting the problems and return to Main Menu

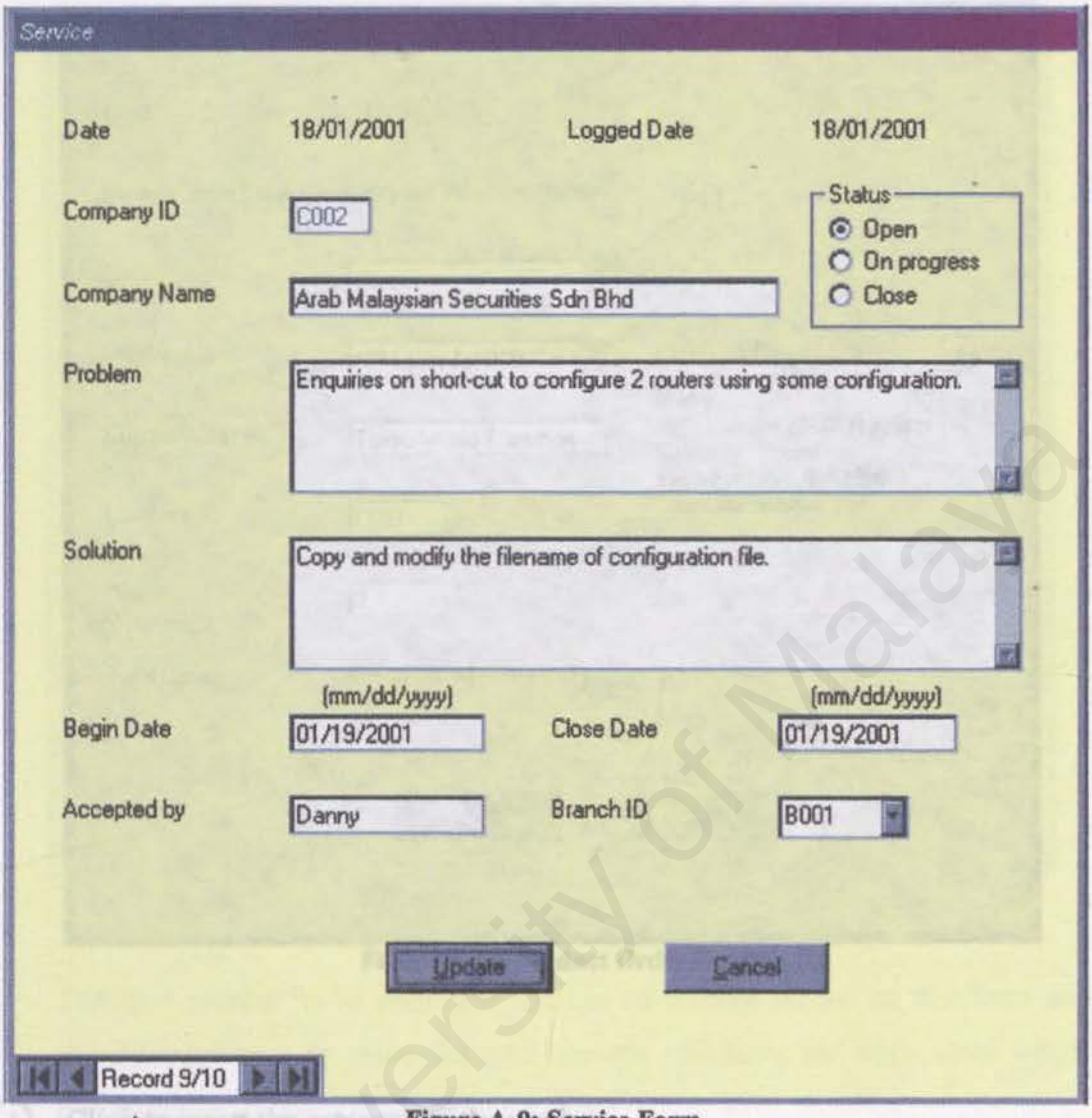


**Figure A-9: Service Form**

***Status Box*** – ***Open*** – Problems that haven't been assign to any engineer to repair

– ***On progress*** – Problems that has been assign an engineer and the repairing is on progress

– ***Close*** – Problems that have been solve

(1) – The first number is to show the number of current record on the form and the second number is to show the total records still have not been close

***Update button*** – To update the database for current record

***Cancel button*** – To close Service From without updating and return to Main Menu

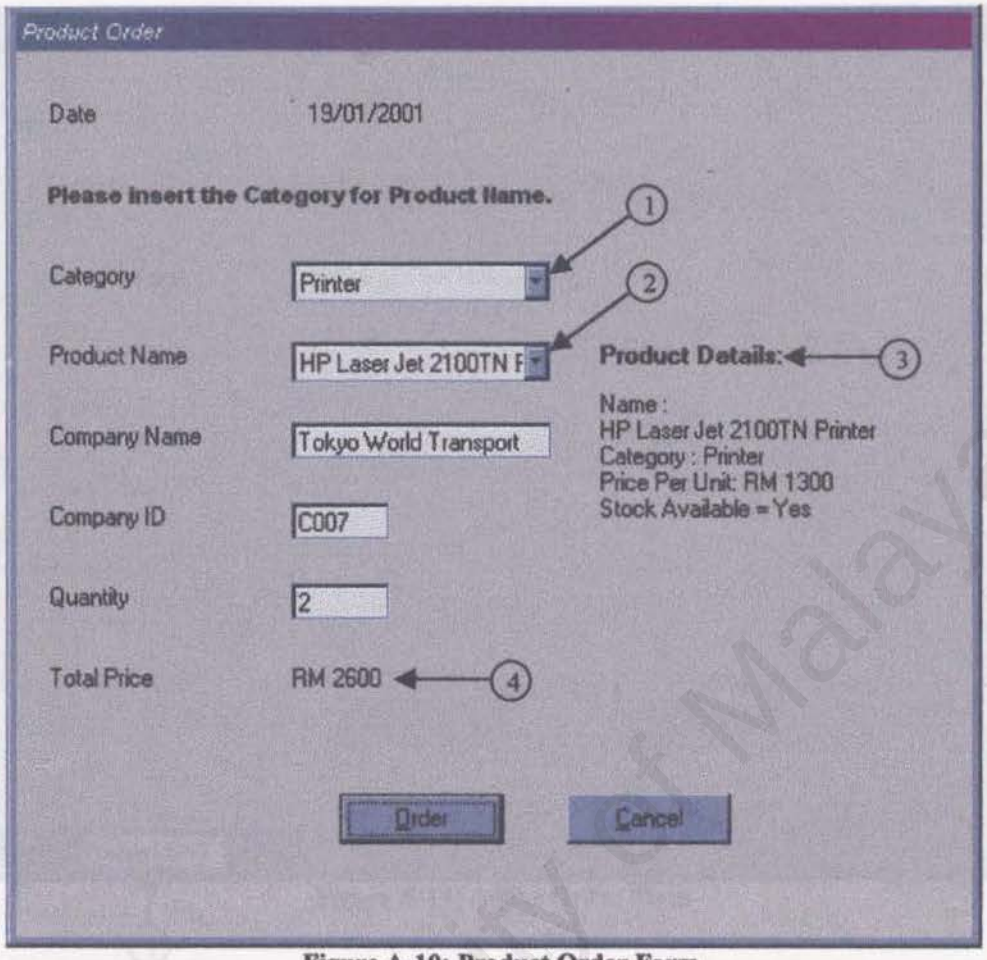


**Figure A-10: Product Order Form**

① – Click to insert the category of the product

② – Click to insert the product name. The product name in this box will appear according to the category insert in ①

③ – The product detail will appear immediately after the product name has been insert

④ – The total price will appear immediately after the quantity of the product to be order have been key in. The total price will appear according to the quantity

***Order button*** – To confirm the product ordering.

***Cancel button*** – To close Product Order From without ordering and return to Main Menu

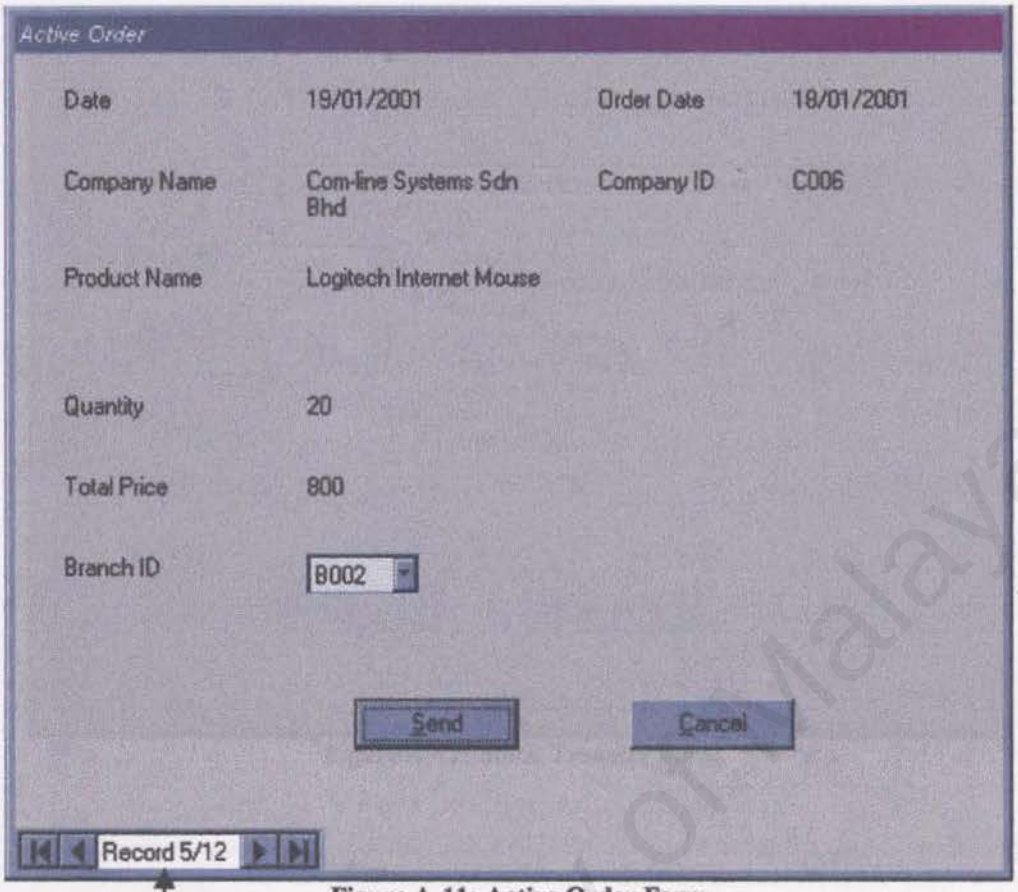


**Figure A-11: Active Order Form**

① – The first number is to show the number of current record on the form and the second number is to show the total records still have not been close where the product have not been send to the customer

***Send button*** – To confirm that the product listed in the current page have been send

***Cancel button*** – To close Active Order From without updating and return to Main Menu

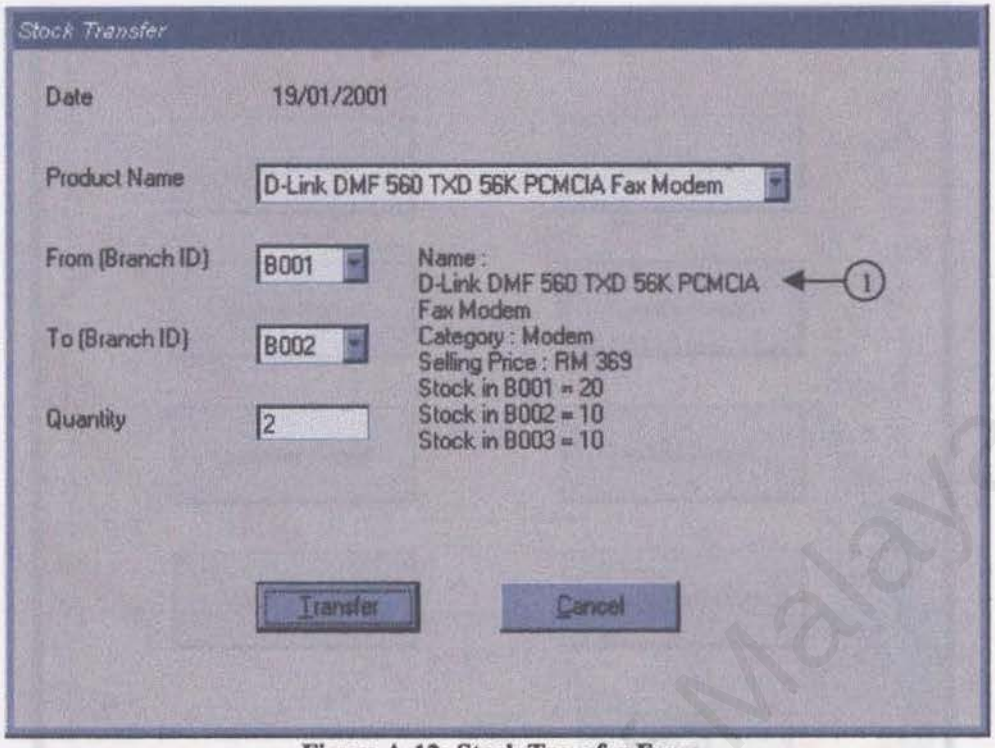


**Figure A-12: Stock Transfer Form**

①– This product information will appear after Product Name is insert

***Transfer button*** – Click to confirm the quantity of product transfer from one branch to another

***Cancel button*** – To close Stock Transfer From without inserting the transfer information into database and return to Main Menu

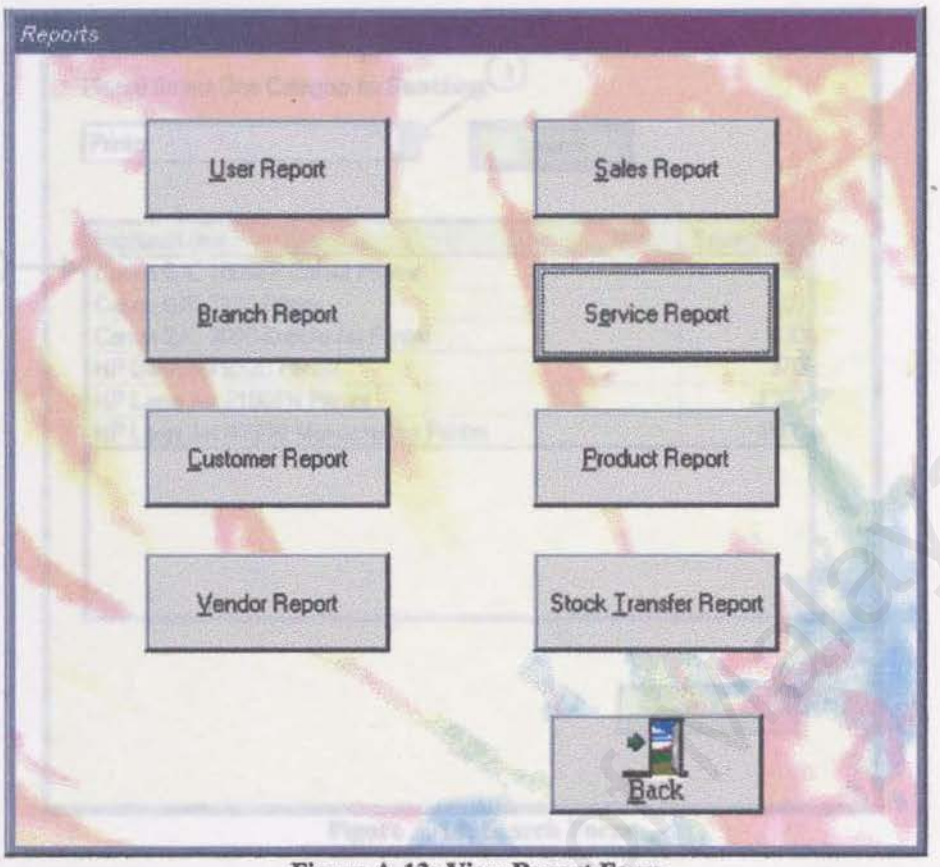


**Figure A-13: View Report Form**

***User Report button*** – Click to view user report

***Branch Report button*** – Click to view branch report

***Customer Report button*** – Click to view customer report

***Vendor Report button*** – Click to view vendor report

***Sales Report button*** – Click to view sales report

***Service Report button*** – Click to view service report

***Product Report button*** – Click to view product report

***Stock Transfer Report button*** – Click to view stock transfer report

***Back button*** – Return to Main Menu

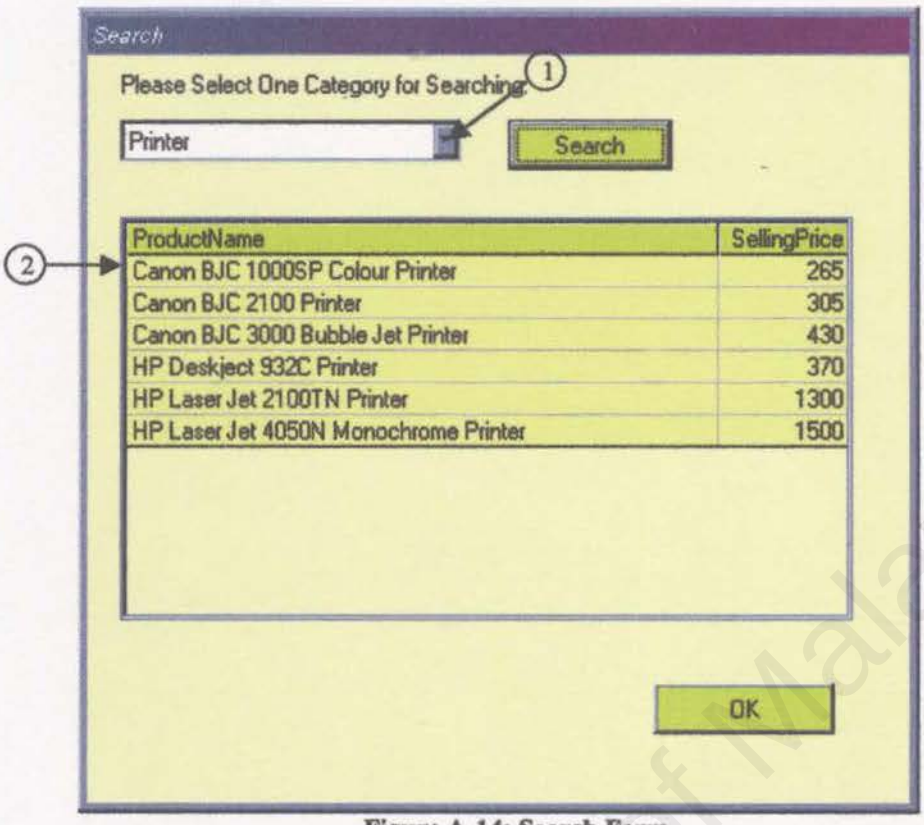


**Figure A-14: Search Form**

① – Click for category of the product

***Search button*** – Click to search according to category insert

② – This box will pop up immediately after Search button is click. This box will pop up according to the category insert

***OK button*** – Return to Main Menu